

Project 1

Building a CI/CD Pipeline for a Retail Company

First of all, thanks very much to edureka and instructors for giving me confidence that I can do any type of devops projects. I just loved working on this project.

I have chosen project 1 out of two projects. So I implemented the 1st project.

github repository:<https://github.com/salvathshaik/final-devops-project.git>

For validation:

please run this pipeline job in jenkins for task-3 deploying to tomcat and docker build and run the container and uploading to dockerhub

CI_CD_DOCKER_V2

please run this pipeline job in jenkins for task-4 deploying to kubernetes and docker using jenkins

CI_CD_PIPELINE_KUBERNETEES_V2/

Business Challenge/Requirement:

ABC Technologies is a leading online retail store, and it has recently acquired a large retail offline business store. The business store has a large number of stores across the globe but is following the conventional pattern of development and deployment. As a result, it has landed at a great loss and is facing the following challenges.

- Low available
- Low scalable
- Low performance
- Hard to built and maintain •
- Developing and deploying are time-consuming

ABC will acquire the data from all these storage systems and plans to use it for analytics and prediction of the firm's growth and sales prospects. In the first phase, ABC has to create the servlets to add a product and display product details. Add servlet dependencies required to compile the servlets. Create an HTML page that will be used to add a product. The team is using Git to keep all the source code.

ABC has decided to use the DevOps model. Once source code is available in GitHub, we need to integrate it with Jenkins and provide continuous build generation for continuous delivery as well as integrate with Ansible and Kubernetes for deployment. Use Docker Hub to pull and push images between Ansible and Kubernetes.

Problem Statements/Tasks:

We need to develop a CI/CD pipeline to automate the software development, testing, packaging, and deployment, reducing the time to market the app and ensuring good quality service is experienced by end users. In this project, we need to—

- push the code to our GitHub repository.
- create a continuous integration pipeline using Jenkins to compile, test, and package the code present in GitHub.
- Write Dockerfile to push the war file to the Tomcat server.
- Integrate Docker with Ansible and write the playbook.
- Deploy artifacts to the Kubernetes cluster • Monitor resources using Grafana.

Approach to Solve:

Task 1: Clone the project from the GitHub link shared in resources to your local machine. Build the code using Maven commands.

Task 2: Set up the Git repository and push the source code. Then, log in to Jenkins.

1. Create a build pipeline containing a job for each
 - One for compiling source code
 - Second for testing source code
 - Third for packing the code
2. Execute the CI/CD pipeline to execute the jobs created in step 1
3. Set up a master-slave node to distribute the tasks in the pipeline

Task 3: Write a Docket file. Create an Image and container on the Docker host. Integrate docker host with Jenkins. Create CI/CD job on Jenkins to build and deploy on a container.

1. Enhance the package job created in step 1 of task 2 to create a docker image.
2. In the Docker image, add code to move the war file to the Tomcat server and build the image.

Task 4: Integrate the Docker host with Ansible. Write an Ansible playbook to create an image and create a continuer. Integrate Ansible with Jenkins. Deploy Ansible-playbook. CI/CD job to build code on ansible and deploy it on docker container

1. Deploy Artifacts on Kubernetes
2. Write pod, service, and deployment manifest file
3. Integrate Kubernetes with Ansible
4. Ansible playbook to create deployment and service

Task 5: Using Prometheus, monitor the resources like CPU utilisation: Total Usage, Usage per core, usage breakdown, memory, and network on the instance by providing the endpoints on the local host. Install the node exporter and add the URL to the target in Prometheus. Using this data, log in to Grafana and create a dashboard to show the metrics.

Let's start...

Task 1: Clone the project from the GitHub link shared in resources to your local machine. Build the code using Maven commands.

Approach i have followed:

- As a given project is based on java I have used maven as a build tool to build the code. maven and java are installed already in the master machine. I have run the maven compile,test, and package tasks successfully in the local environment and produced the results for each and also executed the maven clean install command to build the code.
- Before this I have downloaded the sample Java code from the LMS into my own local environment and used git bash to upload to the below newly created public github repository and also i have uploaded all the code that i have developed to do this project in this repository and all the important files are kept within the directory [project_required_file_v2](#) of project.

Repository: <https://github.com/salvathshaik/final-devops-project.git>

- And executed the below tasks of maven in edureka local environment.
 1. /opt/maven/bin/mvn compile
 2. /opt/maven/bin/mvn test
 3. /opt/maven/bin/mvn package
 4. /opt/maven/bin/mvn clean install
- /opt/maven/bin/mvn - Is the path of the maven executable.
 - 1) /opt/maven/bin/mvn compile: Compiles source code of the project
 - 2) /opt/maven/bin/mvn test: Runs tests for the project.
 - 3) /opt/maven/bin/mvn package: Creates WAR file for the project to convert it into a distributable format.
 - 4) /opt/maven/bin/mvn clean install: Using the **clean** command, which will delete all previously compiled Java .class files and resources (like .properties) in the project. build will start from a clean slate.
- **Install** will then compile, test & package my Java project and even install/copy my built .war file into my local Maven repository.
- Below are the snapshots of results that I did in the edureka provided Lab terminal.

Compile task running in local:

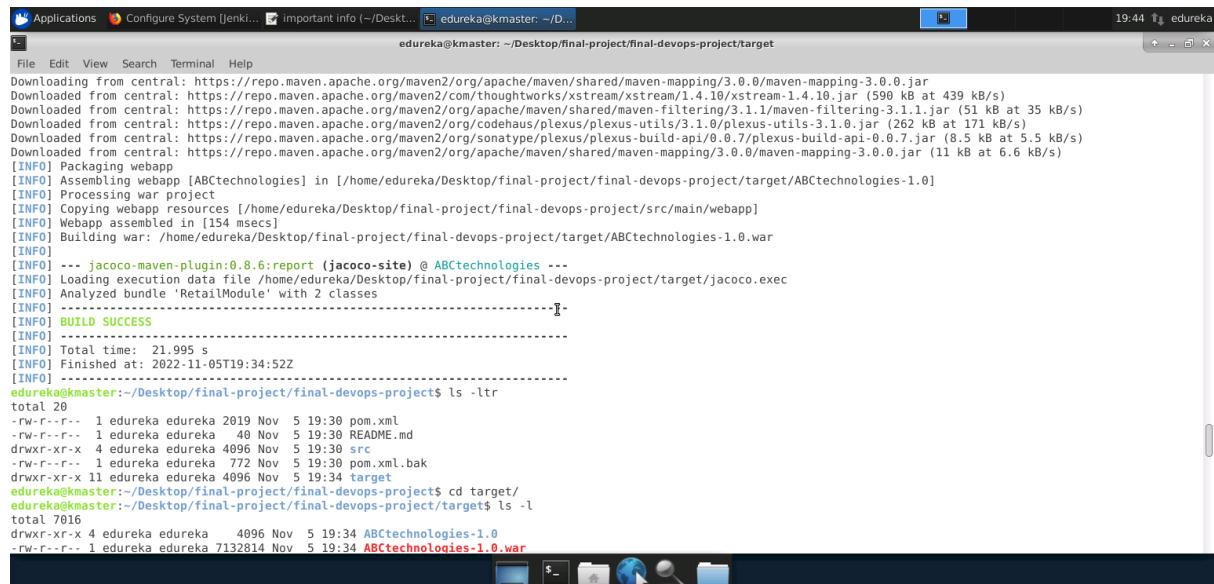
```
edureka@kmaster: ~/Desktop/final-project/final-devops-project/target
File Edit View Search Terminal Help
Downloaded from central: https://repo.maven.apache.org/maven2/org/apache/maven/shared/maven-shared-incremental/1.1/maven-shared-incremental-1.1.jar (14 kB at 8.7 kB/s)
Downloaded from central: https://repo.maven.apache.org/maven2/org/apache/xbean/xbean-reflect/3.4/xbean-reflect-3.4.jar
Downloaded from central: https://repo.maven.apache.org/maven2/org/codehaus/plexus/plexus-compiler-javac/2.2/plexus-compiler-javac-2.2.jar (19 kB at 11 kB/s)
Downloaded from central: https://repo.maven.apache.org/maven2/log4j/1.2.12/log4j-1.2.12.jar
Downloaded from central: https://repo.maven.apache.org/maven2/org/codehaus/plexus/plexus-compiler-manager/2.2/plexus-compiler-manager-2.2.jar (4.6 kB at 2.6 kB/s)
Downloaded from central: https://repo.maven.apache.org/maven2/commons-logging/commons-logging-api/1.1/commons-logging-api-1.1.jar
Downloaded from central: https://repo.maven.apache.org/maven2/org/codehaus/plexus/plexus-classworlds/2.2.2/plexus-classworlds-2.2.2.jar (46 kB at 26 kB/s)
Downloaded from central: https://repo.maven.apache.org/maven2/com/google/collections/1.0/google-collections-1.0.jar
Downloaded from central: https://repo.maven.apache.org/maven2/org/codehaus/plexus/plexus-container-default/1.5.5/plexus-container-default-1.5.5.jar (217 kB at 117 kB/s)
Downloaded from central: https://repo.maven.apache.org/maven2/junit/junit/3.8.2/junit-3.8.2.jar
Downloaded from central: https://repo.maven.apache.org/maven2/org/apache/xbean/xbean-reflect/3.4/xbean-reflect-3.4.jar (134 kB at 70 kB/s)
Downloaded from central: https://repo.maven.apache.org/maven2/commons-logging/commons-logging-api/1.1/commons-logging-api-1.1.jar (45 kB at 21 kB/s)
Downloaded from central: https://repo.maven.apache.org/maven2/log4j/1.2.12/log4j-1.2.12.jar (358 kB at 167 kB/s)
Downloaded from central: https://repo.maven.apache.org/maven2/junit/junit/3.8.2/junit-3.8.2.jar (121 kB at 54 kB/s)
Downloaded from central: https://repo.maven.apache.org/maven2/com/google/collections/google-collections-1.0/google-collections-1.0.jar (640 kB at 282 kB/s)
[INFO] Changes detected - recompiling the module!
[INFO] Compiling 3 source files to /home/edureka/Desktop/final-project/final-devops-project/target/classes
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 01:54 min
[INFO] Finished at: 2022-11-05T19:33:39Z
[INFO] -----
edureka@kmaster:~/Desktop/final-project/final-devops-project$ /opt/maven/bin/mvn test
[INFO] Scanning for projects...
[INFO] -----
< com.abc:ABCtechnologies >
[INFO] Building RetailModule 1.0
[INFO] -----
[INFO] [ war ]
Downloaded from central: https://repo.maven.apache.org/maven2/org/apache/maven/plugins/maven-surefire-plugin/2.12.4/maven-surefire-plugin-2.12.4.pom
Downloaded from central: https://repo.maven.apache.org/maven2/org/apache/maven/plugins/maven-surefire-plugin/2.12.4/maven-surefire-plugin-2.12.4.pom (10 kB at 7.8 kB/s)
Downloaded from central: https://repo.maven.apache.org/maven2/org/apache/maven/surefire/surefire/2.12.4/surefire-2.12.4.pom
Downloaded from central: https://repo.maven.apache.org/maven2/org/apache/maven/surefire/surefire/2.12.4/surefire-2.12.4.pom (14 kB at 35 kB/s)
Downloaded from central: https://repo.maven.apache.org/maven2/org/apache/maven/plugins/maven-surefire-plugin/2.12.4/maven-surefire-plugin-2.12.4.jar
```

Test task running in local:

```
edureka@kmaster: ~/Desktop/final-project/final-devops-project/target
File Edit View Search Terminal Help
Downloaded from central: https://repo.maven.apache.org/maven2/org/apache/maven/plugin-tools/maven-plugin-annotations/3.1/maven-plugin-annotations-3.1.jar (14 kB at 10.0 kB/s)
[INFO] Surefire report directory: /home/edureka/Desktop/final-project/final-devops-project/target/surefire-reports
Downloaded from central: https://repo.maven.apache.org/maven2/org/apache/maven/surefire/surefire-junit4/2.12.4/surefire-junit4-2.12.4.pom
Downloaded from central: https://repo.maven.apache.org/maven2/org/apache/maven/surefire/surefire-junit4/2.12.4/surefire-junit4-2.12.4.pom (2.4 kB at 6.8 kB/s)
Downloaded from central: https://repo.maven.apache.org/maven2/org/apache/maven/surefire/surefire-providers/2.12.4/surefire-providers-2.12.4.pom
Downloaded from central: https://repo.maven.apache.org/maven2/org/apache/maven/surefire/surefire-providers/2.12.4/surefire-providers-2.12.4.pom (2.3 kB at 6.7 kB/s)
Downloaded from central: https://repo.maven.apache.org/maven2/org/apache/maven/surefire/surefire-junit4/2.12.4/surefire-junit4-2.12.4.pom
Downloaded from central: https://repo.maven.apache.org/maven2/org/apache/maven/surefire/surefire-junit4/2.12.4/surefire-junit4-2.12.4.jar (37 kB at 96 kB/s)
-----
T E S T S
-----
Running com.abc.dataAccessObject.ProductImplTest
Tests run: 4, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 0.188 sec
Results :
Tests run: 4, Failures: 0, Errors: 0, Skipped: 0

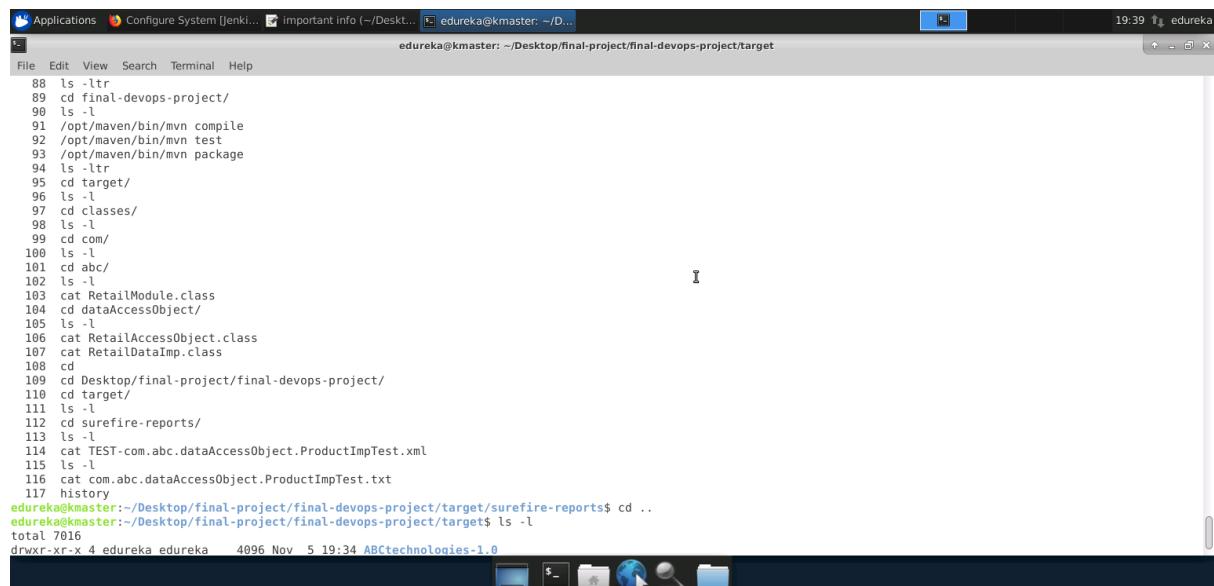
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 15.221 s
[INFO] Finished at: 2022-11-05T19:34:13Z
[INFO] -----
edureka@kmaster:~/Desktop/final-project/final-devops-project$ /opt/maven/bin/mvn package
[INFO] Scanning for projects...
[INFO] -----
< com.abc:ABCtechnologies >
[INFO] Building RetailModule 1.0
[INFO] -----
[INFO] [ war ]
Downloaded from central: https://repo.maven.apache.org/maven2/org/apache/maven/plugins/maven-war-plugin/3.2.2/maven-war-plugin-3.2.2.pom
Downloaded from central: https://repo.maven.apache.org/maven2/org/apache/maven/plugins/maven-war-plugin/3.2.2/maven-war-plugin-3.2.2.pom (9.7 kB at 9.1 kB/s)
```

Package task running in local:



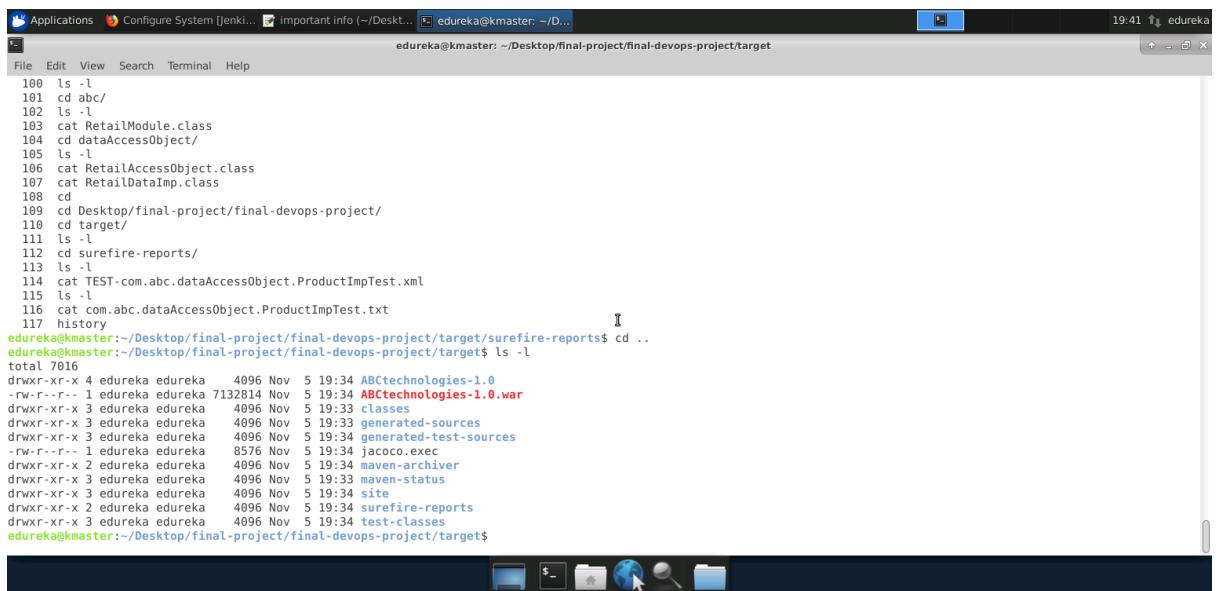
```
edureka@kmaster: ~/Desktop/final-project/final-devops-project/target
File Edit View Search Terminal Help
19:44 edureka
Download from central: https://repo.maven.apache.org/maven2/org/apache/maven/shared/maven-mapping/3.0.0/maven-mapping-3.0.0.jar
Downloaded from central: https://repo.maven.apache.org/maven2/com/thoughtworks/xstream/xstream/1.4.10/xstream-1.4.10.jar (590 kB at 439 kB/s)
Downloaded from central: https://repo.maven.apache.org/maven2/org/apache/maven/shared/maven-filtering/3.1.1/maven-filtering-3.1.1.jar (51 kB at 35 kB/s)
Downloaded from central: https://repo.maven.apache.org/maven2/org/codehaus/plexus/plexus-utils/3.1.0/plexus-utils-3.1.0.jar (262 kB at 171 kB/s)
Downloaded from central: https://repo.maven.apache.org/maven2/org/sonatype/plexus/plexus-build-api/0.0.7/plexus-build-api-0.0.7.jar (8.5 kB at 5.5 kB/s)
Downloaded from central: https://repo.maven.apache.org/maven2/org/apache/maven/shared/maven-mapping/3.0.0/maven-mapping-3.0.0.jar (11 kB at 6.6 kB/s)
[INFO] Packaging webapp
[INFO] Assembling webapp [ABCtechnologies] in [/home/edureka/Desktop/final-project/final-devops-project/target/ABCtechnologies-1.0]
[INFO] Processing war project
[INFO] Copying webapp resources [/home/edureka/Desktop/final-project/final-devops-project/src/main/webapp]
[INFO] Webapp assembled in [154 ms]
[INFO] Building war: /home/edureka/Desktop/final-project/final-devops-project/target/ABCtechnologies-1.0.war
[INFO]
[INFO] --- jacoco-maven-plugin:0.8.6:report (jacoco-site) @ ABCtechnologies ---
[INFO] Loading execution data file /home/edureka/Desktop/final-project/final-devops-project/target/jacoco.exec
[INFO] Analyzed bundle 'RetailModule' with 2 classes
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 21.995 s
[INFO] Finished at: 2022-11-05T19:34:52Z
[INFO] -----
edureka@kmaster:~/Desktop/final-project/final-devops-project$ ls -ltr
total 20
-rw-r--r-- 1 edureka edureka 2019 Nov 5 19:30 pom.xml
-rw-r--r-- 1 edureka edureka 40 Nov 5 19:30 README.md
drwxr-xr-x 4 edureka edureka 4096 Nov 5 19:30 src
-rw-r--r-- 1 edureka edureka 772 Nov 5 19:30 pom.xml.bak
drwxr-xr-x 11 edureka edureka 4096 Nov 5 19:34 target
edureka@kmaster:~/Desktop/final-project/final-devops-project$ cd target/
edureka@kmaster:~/Desktop/final-project/final-devops-project/target$ ls -l
total 7016
drwxr-xr-x 4 edureka edureka 4096 Nov 5 19:34 ABCtechnologies-1.0
-rw-r--r-- 1 edureka edureka 7132814 Nov 5 19:34 ABCtechnologies-1.0.war
```

History and target folder after the maven tasks completed



```
edureka@kmaster: ~/Desktop/final-project/final-devops-project/target
File Edit View Search Terminal Help
19:39 edureka
88 ls -ltr
89 cd final-devops-project
90 ls -l
91 /opt/maven/bin/mvn compile
92 /opt/maven/bin/mvn test
93 /opt/maven/bin/mvn package
94 ls -ltr
95 cd target/
96 ls -l
97 cd classes/
98 ls -l
99 cd com/
100 ls -l
101 cd abc/
102 ls -l
103 cat RetailModule.class
104 cd dataAccessObject/
105 ls -l
106 cat RetailAccessObject.class
107 cat RetailDataImp.class
108 cd
109 cd Desktop/final-project/final-devops-project/
110 cd target/
111 ls -l
112 cd surefire-reports/
113 ls -l
114 cat TEST-com.abc.dataAccessObject.ProductImplTest.xml
115 ls -l
116 cat com.abc.dataAccessObject.ProductImplTest.txt
117 history
edureka@kmaster:~/Desktop/final-project/final-devops-project/target/surefire-reports$ cd ..
edureka@kmaster:~/Desktop/final-project/final-devops-project/target$ ls -l
total 7016
drwxr-xr-x 4 edureka edureka 4096 Nov 5 19:34 ABCtechnologies-1.0
```

Classes folder contains compilation data and package task created .war executable file to deploy.



```
edureka@kmaster: ~/Desktop/final-project/final-devops-project/target
```

```
File Edit View Search Terminal Help
```

```
180 ls -l
181 cd abc/
182 ls -l
183 cat RetailModule.class
184 cd dataAccessObject/
185 ls -l
186 cat RetailAccessObject.class
187 cat RetailDataImp.class
188 cd
189 cd Desktop/final-project/final-devops-project/
190 cd target/
191 ls -l
192 cd surefire-reports/
193 ls -l
194 cat TEST-com.abc.dataAccessObject.ProductImplTest.xml
195 ls -l
196 cat com.abc.dataAccessObject.ProductImplTest.txt
197 history
edureka@kmaster:~/Desktop/final-project/final-devops-project/target$ cd ..
edureka@kmaster:~/Desktop/final-project/final-devops-project/target$ ls -l
total 7016
drwxr-xr-x 4 edureka edureka 4096 Nov 5 19:34 ABCTechnologies-1.0
-rw-r--r-- 1 edureka edureka 7132814 Nov 5 19:34 ABCTechnologies-1.0.war
drwxr-xr-x 3 edureka edureka 4096 Nov 5 19:33 classes
drwxr-xr-x 3 edureka edureka 4096 Nov 5 19:33 generated-sources
drwxr-xr-x 3 edureka edureka 4096 Nov 5 19:34 generated-test-sources
-rw-r--r-- 1 edureka edureka 8578 Nov 5 19:34 jacoco.exec
drwxr-xr-x 2 edureka edureka 4096 Nov 5 19:34 maven-archiver
drwxr-xr-x 3 edureka edureka 4096 Nov 5 19:33 maven-status
drwxr-xr-x 2 edureka edureka 4096 Nov 5 19:34 site
drwxr-xr-x 3 edureka edureka 4096 Nov 5 19:34 surefire-reports
drwxr-xr-x 3 edureka edureka 4096 Nov 5 19:34 test-classes
edureka@kmaster:~/Desktop/final-project/final-devops-project/target$
```

```
19:41 edureka
```

References:

- <https://www.marcobehler.com/guides/mvn-clean-install-a-short-guide-to-maven>
- <https://www.geeksforgeeks.org/maven-lifecycle-and-basic-maven-commands/#:~:text=mvn%20compile%3A%20Compiles%20source%20code,it%20into%20a%20distributable%20format.>

Task 2: Set up the Git repository and push the source code. Then, log in to Jenkins.

2. Create a build pipeline containing a job for each
 - One for compiling source code
 - Second for testing source code
 - Third for packing the code
2. Execute the CI/CD pipeline to execute the jobs created in step 1
3. Set up a master-slave node to distribute the tasks in the pipeline

Approach i have followed:

- Jenkins has been installed in master server already so i have created the 3 jobs for compile,test,package in jenkins and created pipeline with these 3 jobs and set up the agent machine(slave machine) and shared the load to agent as well.

- As a given project is based on java i have used maven to build the code and Jenkins is a build automation server that helps to automate these things so i have set up the java,maven paths of master in global tool configuration in jenkins and set up the jenkins goals and left git path as default.

- Tools location :

`/opt/maven`
`/usr/lib/jvm/java-8-oracle`

- Goals:

compile
test
Package

Please find the screenshots of the above task-2:

Jenkins login:



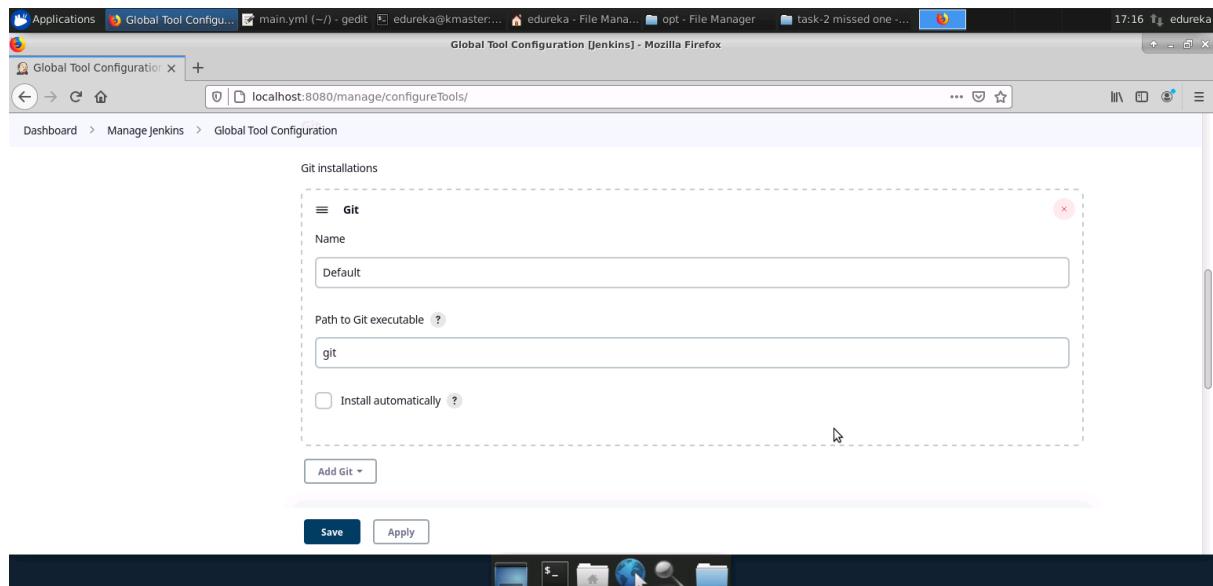
Global tool configuration:

The screenshot shows the Jenkins 'Manage Jenkins' interface. In the top right corner, there is a warning message: "Building on the built-in node can be a security issue. You should set the number of executors on the built-in node to 0. See [the documentation](#)". Below this, under 'System Configuration', there are several links: 'Configure System' (Configure global settings and paths), 'Global Tool Configuration' (Configure tools, their locations and automatic installers), 'Manage Nodes and Clouds' (Add, remove, control and monitor the nodes and clouds used by Jenkins), and 'Manage Plugins' (Add, remove, disable or enable plugins that can extend the functionality of Jenkins). A red box highlights the 'Global Tool Configuration' link.

Tool location configuration for java:

The screenshot shows the Jenkins 'Global Tool Configuration' page for Java. At the top, it says 'JDK installations' and 'List of JDK installations on this system'. There is a button labeled 'Add JDK'. Below it, there is a form for adding a new JDK installation. The 'JDK Name' field contains 'java1.8'. The 'JAVA_HOME' field contains '/usr/lib/jvm/java-8-oracle'. There is also a checkbox labeled 'Install automatically?' which is unchecked. At the bottom of the form are 'Save' and 'Apply' buttons. A red box highlights the 'JAVA_HOME' field.

Tool location configuration for git as default:



The screenshot shows the Jenkins Global Tool Configuration interface for managing Git installations. A single Git installation is defined with the name 'Default' and the path to the executable set to 'git'. There is an unchecked checkbox for 'Install automatically'. Buttons for 'Save' and 'Apply' are at the bottom.

Git installations

Git

Name: Default

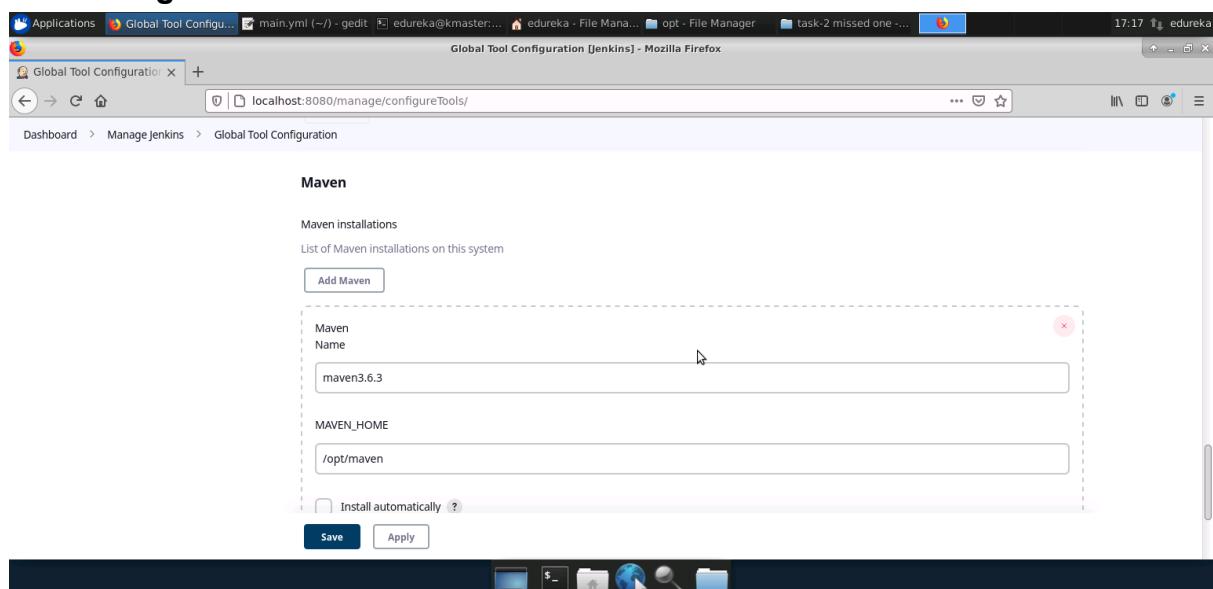
Path to Git executable: git

Install automatically

Add Git

Save Apply

Tool configuration for maven:



The screenshot shows the Jenkins Global Tool Configuration interface for managing Maven installations. A single Maven installation is defined with the name 'maven3.6.3' and the MAVEN_HOME path set to '/opt/maven'. There is an unchecked checkbox for 'Install automatically'. Buttons for 'Save' and 'Apply' are at the bottom.

Maven

Maven installations

List of Maven installations on this system

Add Maven

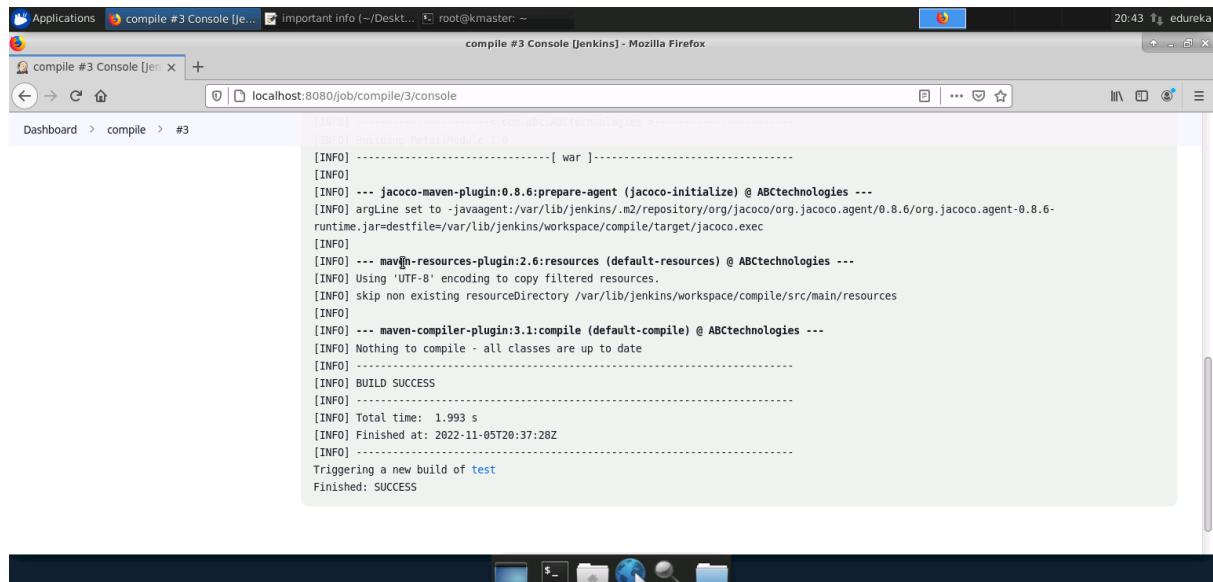
Maven Name: maven3.6.3

MAVEN_HOME: /opt/maven

Install automatically

Save Apply

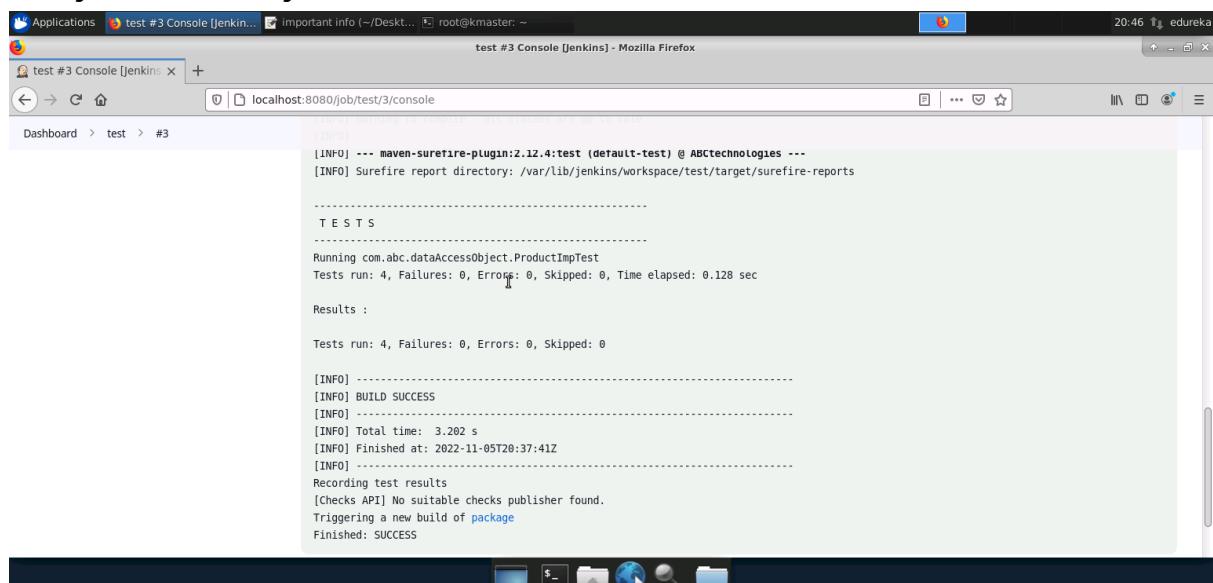
Compile job result in jenkins:



The screenshot shows a Linux desktop environment with a black taskbar at the bottom. On the taskbar, there are icons for Applications, a terminal window titled 'compile #3 Console [Jen...]', and several other application windows. The main focus is a Firefox browser window titled 'compile #3 Console [Jenkins] - Mozilla Firefox'. The address bar shows 'localhost:8080/job/compile/3/console'. The page content displays the Jenkins build log for job 'compile' run #3. The log output is as follows:

```
[INFO] -----[ war ]-----  
[INFO]  
[INFO] --- jacoco-maven-plugin:0.8.6:prepare-agent (jacoco-initialize) @ ABCtechnologies ---  
[INFO] argLine set to -javaagent:/var/lib/jenkins/.m2/repository/org/jacoco/org.jacoco.agent/0.8.6/org.jacoco.agent-0.8.6-  
runtime.jar=destfile=/var/lib/jenkins/workspace/compile/target/jacoco.exec  
[INFO]  
[INFO] --- maven-resources-plugin:2.6:resources (default-resources) @ ABCtechnologies ---  
[INFO] Using 'UTF-8' encoding to copy filtered resources.  
[INFO] skip non existing resourceDirectory /var/lib/jenkins/workspace/compile/src/main/resources  
[INFO]  
[INFO] --- maven-compiler-plugin:3.1:compile (default-compile) @ ABCtechnologies ---  
[INFO] Nothing to compile - all classes are up to date  
[INFO] -----  
[INFO] BUILD SUCCESS  
[INFO] -----  
[INFO] Total time: 1.993 s  
[INFO] Finished at: 2022-11-05T20:37:28Z  
[INFO]  
Triggering a new build of test  
Finished: SUCCESS
```

Test job results in jenkins:



The screenshot shows a Linux desktop environment with a black taskbar at the bottom. On the taskbar, there are icons for Applications, a terminal window titled 'test #3 Console [Jen...]', and several other application windows. The main focus is a Firefox browser window titled 'test #3 Console [Jenkins] - Mozilla Firefox'. The address bar shows 'localhost:8080/job/test/3/console'. The page content displays the Jenkins build log for job 'test' run #3. The log output is as follows:

```
[INFO] --- maven-surefire-plugin:2.12.4:test (default-test) @ ABCtechnologies ---  
[INFO] Surefire report directory: /var/lib/jenkins/workspace/test/target/surefire-reports  
-----  
T E S T S  
-----  
Running com.abc.dataAccessObject.ProductImpTest  
Tests run: 4, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 0.128 sec  
Results :  
Tests run: 4, Failures: 0, Errors: 0, Skipped: 0  
-----  
[INFO] BUILD SUCCESS  
-----  
[INFO] Total time: 3.202 s  
[INFO] Finished at: 2022-11-05T20:37:41Z  
[INFO]  
Recording test results  
[Checks API] No suitable checks publisher found.  
Triggering a new build of package  
Finished: SUCCESS
```

test [Jenkins] - Mozilla Firefox

localhost:8080/job/test/

Jenkins

Dashboard > test >

Project test

[Back to Dashboard](#)

Status

- </> Changes
- Workspace
- Build Now
- Configure
- Delete Project
- Rename

[Add description](#) [Disable Project](#)

Upstream Projects

Latest Test Result (no failures)

Downstream Projects

compile package

Build History [trend](#)

Permalinks

[Dashboard](#) [test](#) [#3](#) [Test Results](#)

test #3 Test Results [Jenkins]

localhost:8080/job/test/3/testReport/

Test Result

0 failures (± 0)

4 tests (± 0) Took 20 ms.

[Add description](#)

All Tests

Package	Duration	Fail (diff)	Skip (diff)	Pass (diff)	Total (diff)
com.abc.dataAccessObject	20 ms	0	0	4	4

[REST API](#) [Jenkins 2.361.1](#)

[Dashboard](#) [test](#) [#3](#) [Test Results](#)

Package job result in jenkins:

package #3 Console [Jenkins]

localhost:8080/job/package/3/console

Results :

```

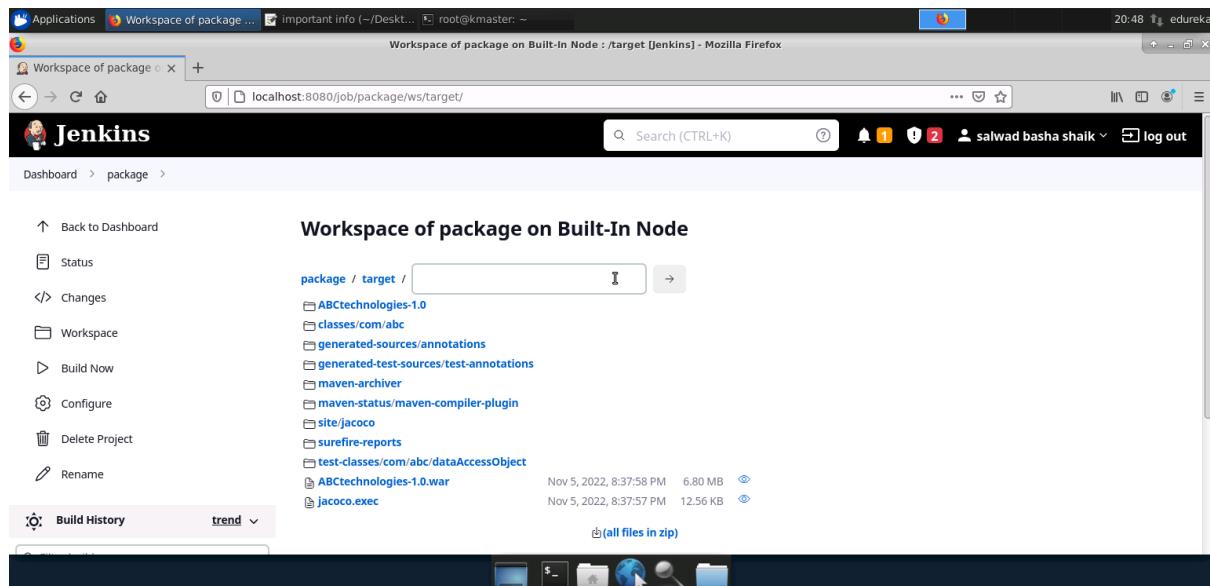
Tests run: 4, Failures: 0, Errors: 0, Skipped: 0

[INFO]
[INFO] --- maven-war-plugin:3.2.2:war (default-war) @ ABCtechnologies ---
[INFO] Packaging webapp
[INFO] Assembling webapp [ABCtechnologies] in [/var/lib/jenkins/workspace/package/target/ABCtechnologies-1.0]
[INFO] Processing war project
[INFO] Copying webapp resources [/var/lib/jenkins/workspace/package/src/main/webapp]
[INFO] Webapp assembled in [91 msecs]
[INFO] Building war: [/var/lib/jenkins/workspace/package/target/ABCtechnologies-1.0.war
[INFO]
[INFO] --- jacoco-maven-plugin:0.8.6:report (jacoco-site) @ ABCtechnologies ---
[INFO] Loading execution data file [/var/lib/jenkins/workspace/package/target/jacoco.exec
[INFO] Analyzed bundle 'RetailModule' with 2 classes
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 5.051 s
[INFO] Finished at: 2022-11-05T20:37:58Z
[INFO] -----
Finished: SUCCESS

```

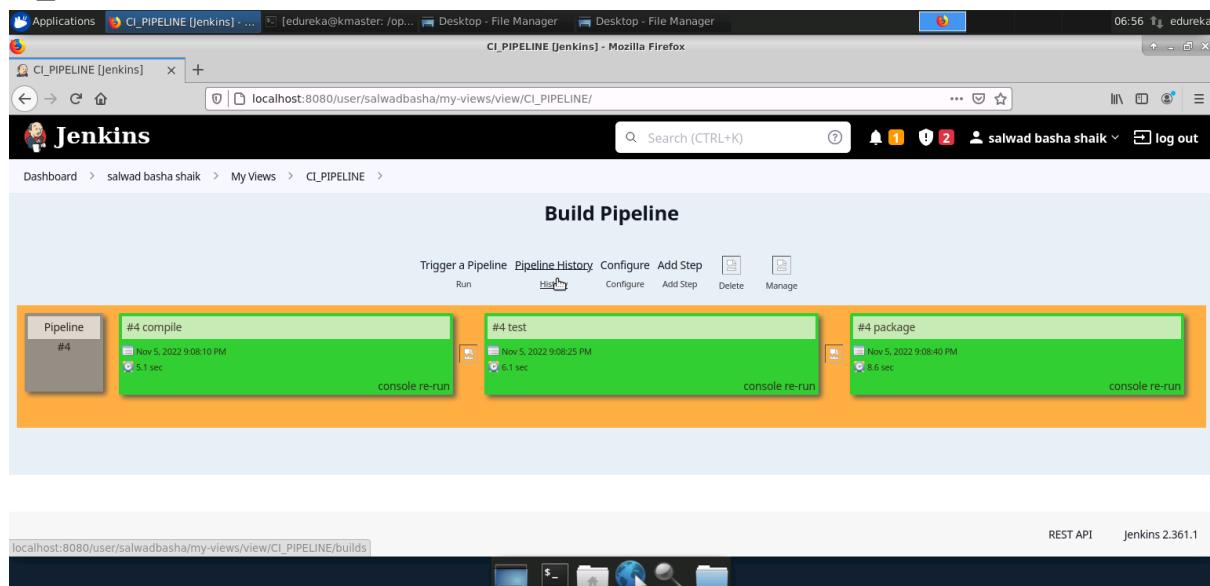
[Dashboard](#) > package > #3

Target folder in jenkins:



Step 2: Execute the CI/CD pipeline to execute the jobs created in step 1

CI_PIPELINE:



And I have configured the git polling as a **cron job** for a compile job that triggers for every 2 minutes for github code so that if whatever change happens then it will trigger and perform the given task.

<https://github.com/salvathshaik/final-devops-project.git>

Poll SCM settings using cronjob in compile job:

The screenshot shows the Jenkins job configuration page for a job named 'compile'. Under the 'Build Triggers' section, the 'Poll SCM' option is selected with a cron schedule of 'H/2 * * * *'. A note below indicates the next run will be at Saturday, November 5, 2022, at 9:04:19 PM Coordinated Universal Time. There is also an 'Ignore post-commit hooks' checkbox. At the bottom are 'Save' and 'Apply' buttons, and a search bar containing 'kolka'.

Git polling Log that it has triggered automatically after i modified the Readme file.

The screenshot shows the Jenkins 'Git Polling Log' page for the same 'compile' job. The log output details the git polling process, starting on Nov 5, 2022, at 9:08:00 PM. It shows the strategy used, the last built revision, and the command executed to find changes. The log concludes with 'Changes found'. At the bottom is a search bar containing 'kolka'.

```
Started on Nov 5, 2022, 9:08:00 PM
Using strategy: Default
[poll] Last Built Revision: Revision b72dcf188dbec71ce0006312991929cb757f9730 (refs/remotes/origin/main)
The recommended git tool is: NONE
No credentials specified
> git --version # timeout=10
> git --version # 'git version 2.17.1'
> git ls-remote -h -- https://github.com/salvathshaik/final-devops-project.git # timeout=10
Found 1 remote heads on https://github.com/salvathshaik/final-devops-project.git
[poll] Latest remote head revision on refs/heads/main is: 5cb2dc208d6cc323a8c3b7cf36983b2a3b0a6737
Done. Took 0.35 sec
Changes found
```

Step 3: Set up a master-slave node to distribute the tasks in the pipeline

The agent configurations:

- I have created a **user jenkins** on second linux machine so that i can run the jenkins jobs in slave machine as well.
- When i try to configure the agent i was getting compatibility issue while running the agent command in agent server(slave server) and got to know that this is due to java 8 compatibility issue beginning with Jenkins 2.357 and Jenkins 2.361.1, running the controller on Java 11 and agents on Java 8 will result in the following error:

```
Error: A JNI error has occurred, please check your installation and try again
Exception in thread "main" java.lang.UnsupportedClassVersionError:
hudson/remoting/Launcher has been compiled by a more recent version of
the Java Runtime (class file version 55.0), this version of the Java
Runtime only recognizes class file versions up to 52.0
        at java.lang.ClassLoader.defineClass1(Native Method)
        at java.lang.ClassLoader.defineClass(ClassLoader.java:756)
        at
java.security.SecureClassLoader.defineClass(SecureClassLoader.java:142)
        at java.net.URLClassLoader.defineClass(URLClassLoader.java:473)
        at java.net.URLClassLoader.access$100(URLClassLoader.java:74)
        at java.net.URLClassLoader$1.run(URLClassLoader.java:369)
        at java.net.URLClassLoader$1.run(URLClassLoader.java:363)
        at java.security.AccessController.doPrivileged(Native Method)
        at java.net.URLClassLoader.findClass(URLClassLoader.java:362)
        at java.lang.ClassLoader.loadClass(ClassLoader.java:418)
        at sun.misc.Launcher$AppClassLoader.loadClass(Launcher.java:352)
        at java.lang.ClassLoader.loadClass(ClassLoader.java:351)
        at
sun.launcher.LauncherHelper.checkAndLoadMain(LauncherHelper.java:601)
```

So I installed the **java 11** version in the agent machine and set up the environment variables to make the java 11 as default version in both jenkins and agent machines.

```
export JAVA_HOME=/usr/lib/jvm/java-11-openjdk-amd64
export PATH=$JAVA_HOME/bin:$PATH
```

After upgrading the java8 version to java11 i am able to run the below agent commands in agent machine and got connected to the agent from jenkins console and able to run the jobs in agent machine and also configured the compile job to run this job only in agent machine.

I have created a **/opt/jagenthome** directory in the agent machine so that it will be used as a working directory to put the result and I have set the jenkins url with the private ip of the master node because if it is with the public ip it will always changes after every restart.

Agent commands that i run in agent machine:

```
> curl -sO http://172.31.10.56:8080/jnlpJars/agent.jar
> java -jar agent.jar -jnlpUrl http://172.31.10.56:8080/computer/agent01/jenkins-agent.jnlp
-secret bd3430016519c105e23863c50b9f01057092154668f863d217a035bbe7e79d2c
-workDir "/opt/jagenthome"
```

After running ...

Agent connected validation in jenkins

The screenshot shows the Jenkins interface for managing agents. The title bar says "agent01 [Jenkins] - Mozilla Firefox". The main content area is titled "Agent agent01". It displays the message "Agent is connected." and lists "Projects tied to agent01" as "None". On the left, there's a sidebar with options like "Status", "Delete Agent", "Configure", "Build History", "Load Statistics", "Script Console", "Log", "System Information", and "Disconnect". A "Mark this node temporarily offline" button is located in the top right. The bottom of the screen shows a toolbar with icons for desktop, file manager, search, and other tools.

Package job running in agent01 machine(example)

The screenshot shows the Jenkins dashboard under a user's view. The title bar says "Dashboard [Jenkins] - Mozilla Firefox". The main content area is titled "CI_PIPELINE". It shows a table with columns "S", "W", "Name", "Last Success", "Last Failure", and "Last Duration". The table contains three rows: "compile" (last success 2 min 35 sec, last failure N/A, duration 1 min 48 sec), "package" (last success 9 hr 50 min, last failure N/A, duration 8.6 sec), and "test" (last success 40 sec, last failure N/A, duration 18 sec). To the left, there are sections for "Build Queue" (empty), "Build Executor Status" (listing "agent01" with status "idle"), and "Built-In Node" (listing "agent01" with status "idle"). At the bottom, there are links for "Icon legend", "Atom feed for all", "Atom feed for failures", and "Atom feed for just latest builds". The address bar at the bottom shows "localhost:8080/user/salwadasha/my-views/view/all/job/package/lastSuccessfulBuild/". The bottom of the screen shows a toolbar with icons for desktop, file manager, search, and other tools.

Making a compile job to run in agent01 machine only.

The screenshot shows the Jenkins configuration page for a job named 'compile'. In the 'General' section, under 'Restrict where this project can be run', the 'Label Expression' field contains 'agent01'. A tooltip indicates that 'Label agent01 matches 1 node. Permissions or other restrictions provided by plugins may further reduce that list.' Below this, the 'Source Code Management' section is set to 'None'. At the bottom, there are 'Save' and 'Apply' buttons.

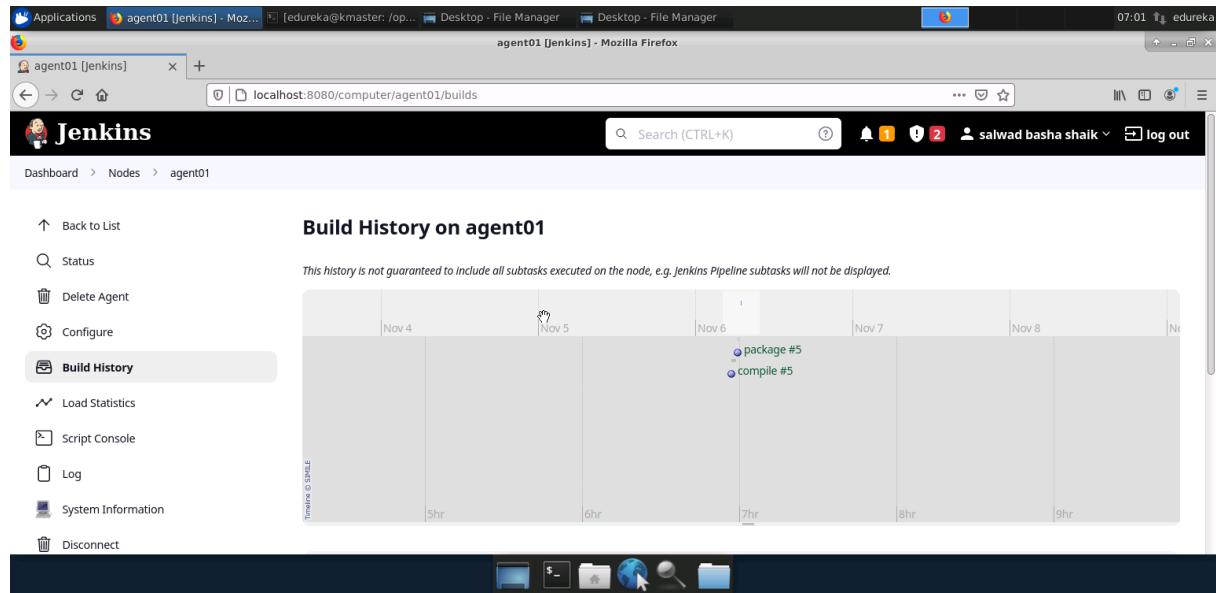
Compile job using /opt/jagenthome page as workspace

The screenshot shows the Jenkins console output for build #9. The title bar says 'compile #9 Console [jenkins] - Mozilla Firefox'. The main content area shows the build log:

```
Started by user salwad_basha_shaiik
Running as SYSTEM
Building remotely on agent01 in workspace /opt/jagenthome/workspace/compile
The recommended git tool is: NONE
No credentials specified
> /usr/bin/git rev-parse --resolve-git-dir /opt/jagenthome/workspace/compile/.git # timeout=10
Fetching changes from the remote Git repository
> /usr/bin/git config remote.origin.url https://github.com/salvathshaik/final-devops-project.git # timeout=10
Fetching upstream changes from https://github.com/salvathshaik/final-devops-project.git
> /usr/bin/git --version # timeout=10
> git --version # 'git' version 2.17.1'
> /usr/bin/git fetch --tags --progress -- https://github.com/salvathshaik/final-devops-project.git +refs/heads/*:refs/remotes/origin/* # timeout=10
> /usr/bin/git rev-parse refs/remotes/origin/main^{commit} # timeout=10
Checking out Revision 5cb2dc200d6cc323a8c3b7cf36983b2a3b0a6737 (refs/remotes/origin/main)
> /usr/bin/git config core.sparsecheckout true
> /usr/bin/git checkout -f 5cb2dc200d6cc323a8c3b7cf36983b2a3b0a6737
Branch 'main' set up to track remote branch 'main' from 'origin'.

```

Agent01 build history



Note: I have modified the compile job again for the next set of tasks.

References:

- <https://www.jenkins.io/blog/2022/06/28/require-java-11/>
- <https://www.jenkins.io/doc/administration/requirements/upgrade-java-guidelines/#:~:text=If%20you're%20upgrading%20your,screen%20of%20your%20Jenkins%20instance.>
- <https://stackoverflow.com/questions/69495517/unable-to-install-jenkins-on-ubuntu-20-04>
- <https://stackoverflow.com/questions/14119983/java-home-and-path-are-set-but-java-version-still-shows-the-old-one>

Task 3: Write a Docket file. Create an Image and container on the Docker host. Integrate docker host with Jenkins. Create CI/CD job on Jenkins to build and deploy on a container.

1. Enhance the package job created in step 1 of task 2 to create a docker image.
2. In the Docker image, add code to move the war file to the Tomcat server and build the image

Approach i have followed:

- Jenkins and Docker are already installed in master machine and slave machine.
- Now using earlier jenkins package job with more enhancement with docker integration and doing two things here
 - i) Deploying .war file generated from package job into the tomcat server
 - ii) And creating the docker build and docker container with above .war file generated from package command and uploading the docker image to the docker hub and running this docker image as container.

Step1: Deploying .war file generated from package job into the tomcat server

Issue 1:

And as part of Tomcat deployment I have tried to deploy using shell commands by copying the .war file to the webapps/ folder but it did not work and I got the below error at that time.

Error message:

```
"sudo: no tty present and no askpass program specified  
Build step 'Execute shell' marked build as failure  
Finished: FAILURE"
```

I thought to copy the SSH public file to the slave machine but it not worked out. So finally I followed a new way to deploy the .war file.

Solution:

Added the 'Deploy to container' plugin and modified the **package_docker_v2** job and configured the **post-build** action to deploy to the tomcat server.

Issue 2:

After running the job I got an issue and was not able to get into the tomcat so i have configured the tomcat **conf/tomcat-users.xml** file with the below addition to the end like i have created a user as a tomcat and password as a admin for the role of manager-gui and manager_script and later it allowed me to deployed to the tomcat successfully.

Solution:

```
conf/tomcat-users.xml  
<role rolename="manager-gui"/>  
<role rolename="manager-script"/>  
<user username="tomcat" password="admin" roles="manager-gui,manager-script"/>
```

Issue 3:

While running the build i got issue that jenkins not able to deploying to the tomcat and figure out that whenever we restarted Lab tomcat will not start automatically so i have added one more build stage as Execute shell to start the tomcat and also got error even after this because tomcat will not initialize immediate start so i have included sleep 10 command as well for it to initialize totally.

Solution:

```
> sudo bash /opt/tomcat/bin/startup.sh  
> sleep 10
```

Step 2: Creating the docker build and docker container with above .war file generated from package command and uploading the docker image to the docker hub and running this docker image as container.

- As part of building docker images and uploading to dockerhub and running in the docker container.

I tried to build the docker image manually in local and it got successful as part of that i have created a **Dockerfile** and added the plugins "cloudbees docker build and publish", "docker pipeline", "docker plugin", "docker build step".

After installation of plugins you should be able to see the build/publish docker image in the build section of jenkins job and one more option "docker build and publish" , if you see that your plugin is installed successfully.

For telling jenkins to prepare docker image we need to build the DockerFile , now we need to give instructions to jenkins but it will read this Dockerfile in the home directory so i have placed this file in home directory and after build i have pushed the docker image to the **docker hub** for that i need to give my docker hub credentials to jenkins so for giving credentials i have configured the job in the build section "**Docker build and publish**" and validated in local and docker hub after build the job.

And while running the job i get into the below issues

Issue 4:

Got permission denied while trying to connect to the Docker daemon socket at unix:///var/run/docker.sock: Get http://%2Fvar%2Frun%2Fdocker.sock/v1.40/containers/json: dial unix /var/run/docker.sock: connect: permission denied

Solution:

I have added the below commands in newly created build section Execute shell before starting to build the docker image

```
sudo chmod 666 /var/run/docker.sock  
sudo chmod 777 /var/lib/jenkins/.docker/config.json
```

Sudo chmod 777 .docker

And also added the below additions in /etc/sudoers file(sudo file)

jenkins ALL=(ALL) NOPASSWD: ALL

Please find the screenshots:

Issue 5:

jenkins.model.InvalidBuildsDir: \${ITEM_ROOTDIR}/builds does not exist and probably cannot be created at jenkins.model.Jenkins.checkRawBuildsDir(Jenkins.java:3085) at jenkins.model.Jenkins.loadConfig(Jenkins.java:3009) Caused: java.io.IOException

Solution:

experienced this when the \$JENKINS_HOME/jobs directory had incorrect permissions or ownership so provided the permissions for jenkins user for this jenkins home directory.

Issue 6:

While building docker image with Dockerfile it is not recognizing the .war file path that i have mentioned in Dockerfile and got below error

```
((ADD /var/lib/jenkins/workspace/package_kubernetes2/target/ABCtechnologies-1.0.war  
/usr/local/tomcat/webapps ADD failed: file not found in build context or excluded by  
.dockerignore: stat  
var/lib/jenkins/workspace/package_kubernetes2/target/ABCtechnologies-1.0.war))
```

Solution:

So i did modified the Dockerfile with the below changes **/*.war instead of target/*.war

Steps to configure the job for docker integration after installed above mentioned plugins Create build tab with maven package and then 'docker build and publish' plugin added and go to configure In build tab -> select 'docker build and publish' -> Repository name(**salvathshaik/abctechnologies**) -> and add Registry credentials with '**username and password**'- and that's it it will get success.

Dockerfile

```
FROM docker.io/library/ubuntu:18.04  
RUN apt-get -y update && apt-get -y upgrade  
RUN apt-get -y install openjdk-8-jdk wget  
RUN mkdir /usr/local/tomcat  
ADD https://dlcdn.apache.org/tomcat/tomcat-9/v9.0.68/bin/apache-tomcat-9.0.68.tar.gz  
/tmp/apache-tomcat-9.0.68.tar.gz  
RUN cd /tmp && tar xvfz apache-tomcat-9.0.68.tar.gz  
RUN cp -Rv /tmp/apache-tomcat-9.0.68/* /usr/local/tomcat/  
ADD **/*.war /usr/local/tomcat/webapps  
EXPOSE 8080  
CMD /usr/local/tomcat/bin/catalina.sh run
```

Below are the screenshots of the above 2 steps of task-3

Package_docker_v2 job settings

Git repo settings1

The screenshot shows the Jenkins job configuration page for 'package_docker_v2'. The left sidebar has 'Source Code Management' selected under 'Configuration'. Under 'Source Code Management', 'Git' is selected. In the main panel, 'Repositories' is set to 'https://github.com/salvathshailk/final-devops-project.git'. The 'Credentials' dropdown is set to '- none -'. There is an 'Add' button and an 'Advanced...' link. Below the repository section is a 'Branches to build' section with a dropdown menu containing '^main'. At the bottom are 'Save' and 'Apply' buttons.

Git repo settings1

The screenshot shows the Jenkins job configuration page for 'package_docker_v2'. The left sidebar has 'Source Code Management' selected under 'Configuration'. Under 'Source Code Management', 'Git' is selected. In the main panel, 'Branches to build' is set to 'Branch Specifier (blank for any)': '^main'. The 'Repository browser' dropdown is set to '(Auto)'. At the bottom are 'Save' and 'Apply' buttons.

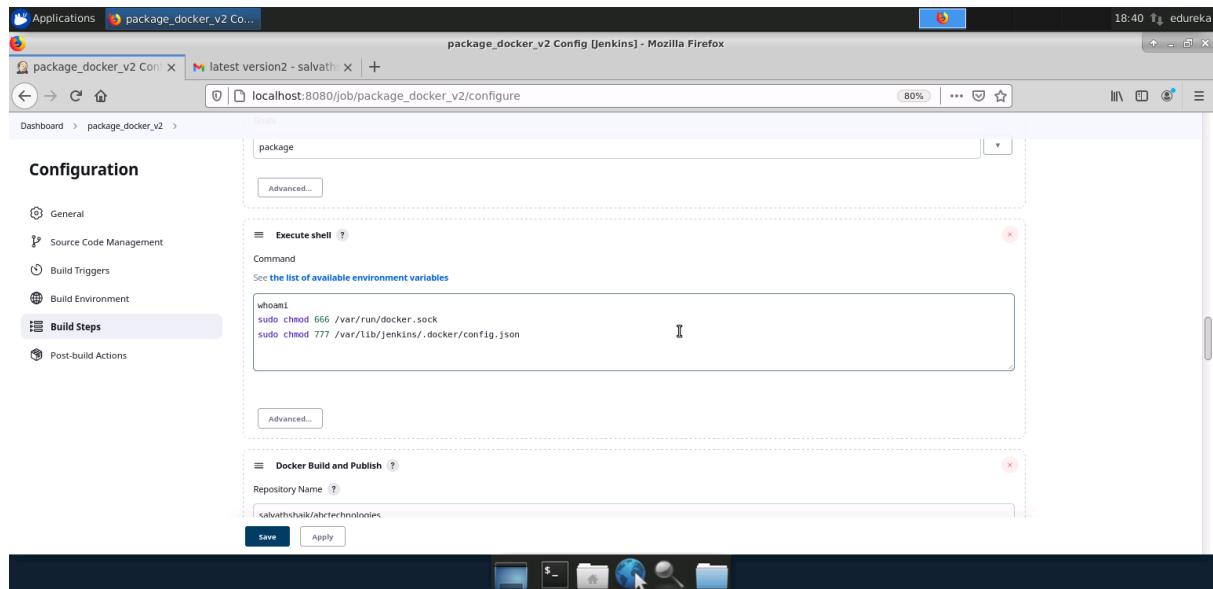
Choosing 'Invoke top-level Maven targets' option to add maven goal

The screenshot shows the Jenkins configuration interface for the 'package_docker_v2' job. The 'Post-build Actions' section is selected. A dropdown menu titled 'Add build step' is open, showing various options. The 'Invoke top-level Maven targets' option is highlighted. At the bottom of the screen, there are several icons for navigating between tabs and performing common tasks.

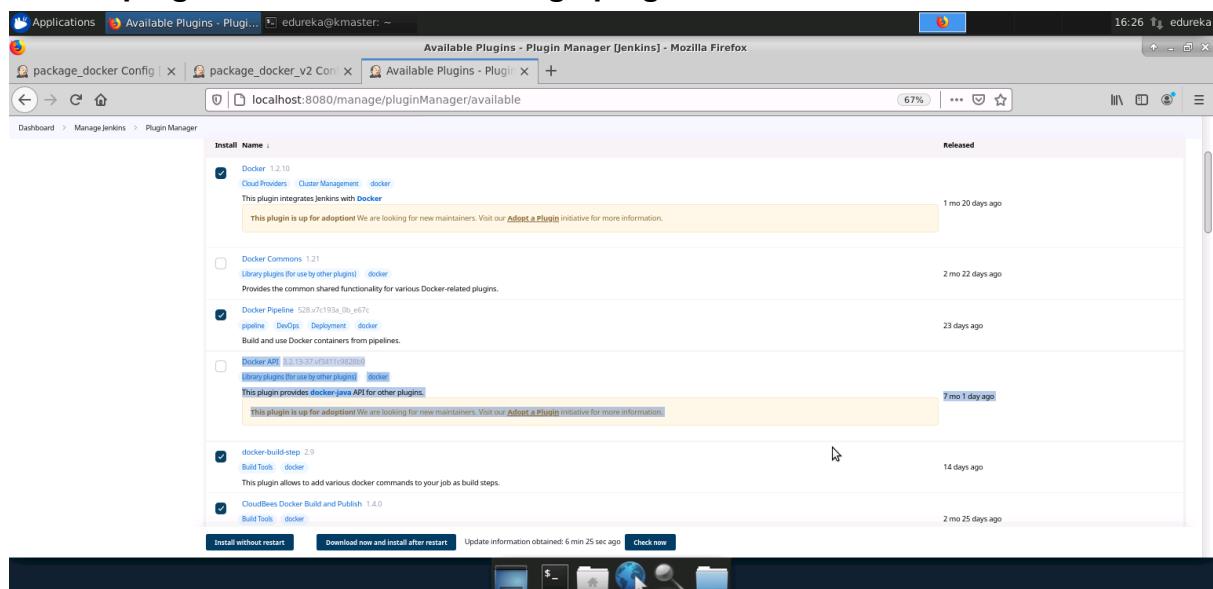
Maven setting1

The screenshot shows the Jenkins configuration interface for the 'package_docker_v2' job. The 'Build Steps' section is selected. It contains three build steps: 1) 'Invoke top-level Maven targets' with Maven Version set to 'maven3.6.3' and Goals set to 'package'. 2) 'Execute shell' with the command: 'whoami
sudo chmod 666 /var/run/docker.sock'. At the bottom of the screen, there are several icons for navigating between tabs and performing common tasks.

Giving permissions to the jenkins user by adding below commands in Execute shell in build step.



Docker plugins installed from manage plugins->available tab



Docker plugins installed successfully

The screenshot shows the Jenkins Update Center interface. At the top, there are tabs for 'package_docker Config' and 'package_docker_v2 Config'. The main content area displays a list of installed plugins under 'Installing Plugins/Upgrades'. Most entries show a green checkmark and the word 'Success'. The listed plugins include:

- Authentication Tokens API
- Docker Commons
- Docker API
- Docker
- Docker Pipeline
- Maven Integration
- docker-build-step
- CloudBees Docker Build and Publish
- Loading plugin extensions

Below the list, there are two links: 'Go back to the top page' and 'Restart Jenkins when installation is complete and no jobs are running'.

'Docker build and publish' build step settings

The screenshot shows the Jenkins job configuration for 'package_docker_v2'. The left sidebar has a 'Build Steps' section selected. The main panel shows the 'Docker Build and Publish' configuration screen. The 'Repository Name' field contains 'salvathshaik/abctechnologies'. The 'Tag' field is empty. The 'Docker Host URI' field is empty. Under 'Server credentials', a dropdown menu shows '- none -'. The 'Docker registry URL' field is empty. At the bottom, there are 'Save' and 'Apply' buttons.

Click on Add credentials and give user name and password for docker login

The image consists of two vertically stacked screenshots of a Jenkins job configuration page. Both screenshots show the 'Post-build Actions' section selected on the left sidebar.

Screenshot 1 (Top): Adding a Jenkins Credential Provider

In this screenshot, a modal window titled 'Jenkins Credentials Provider: Jenkins' is open. It shows the 'Add Credentials' form. The 'Domain' dropdown is set to 'Global credentials (unrestricted)'. The 'Kind' dropdown is set to 'Username with password'. The 'Scope' dropdown is set to 'Global (Jenkins, nodes, items, all child items, etc.)'. The 'Username' field contains 'salvathshaik'. At the bottom of the modal are 'Save' and 'Apply' buttons.

Screenshot 2 (Bottom): Completing the Credential Details

In this screenshot, the same modal window is shown, but the 'Kind' dropdown has been changed to 'Password'. The 'Username' field still contains 'salvathshaik'. A new 'ID' field is present, containing 'DOCKERHUB_CREDS'. The 'Description' field also contains 'DOCKERHUB_CREDS'. At the bottom of the modal are 'Add' and 'Cancel' buttons, along with 'Save' and 'Apply' buttons at the bottom of the main configuration page.

After setting above credentials selecting the credentials in 'Registry credentials'

The screenshot shows the Jenkins configuration interface for a job named 'package_docker_v2'. The 'Post-build Actions' section is selected. It contains a 'Docker registry URL' field and a 'Registry credentials' dropdown set to 'salvathshaik***** (DOCKERHUB_CREDS)'. Below this is an 'Add build step' button. The 'Post-build Actions' section includes an 'Add post-build action' dropdown and buttons for 'Save' and 'Apply'. The top navigation bar shows tabs for 'package_docker_v2 Config' and 'localhost:8080/job/package_docker_v2/configure'.

After pushing the image to dockerhub starting the docker container

The screenshot shows the Jenkins configuration interface for the same job. The 'Post-build Actions' section now contains two 'Execute shell' steps. The first step's command is:

```
echo "Successfully build and uploaded to dockerhub"  
sudo docker run -d -P salvathshaik/abctechnologies  
echo "Successfully deployed to the docker container by pulling the image from dockerhub"
```

The second step's command is:

```
curl http://localhost:8080/tomcat/tomcat8/startup.sh
```

Below the actions are 'Advanced...' and 'Save' buttons. The top navigation bar shows tabs for 'package_docker_v2 Config' and 'localhost:8080/job/package_docker_v2/configure'.

Before deploying to tomcat, I am starting the tomcat and waiting for sleep 10 seconds to initialize completely.

The screenshot shows the Jenkins configuration interface for the 'package_docker_v2' job. The 'Post-build Actions' section is selected. A 'Execute shell' action is defined with the command:

```
sudo docker run -d -P salvathshaik/abctechnologies
echo "Successfully deployed to the docker container by pulling the image from dockerhub"
```

Selecting 'Deploy war/ear to a container' for tomcat settings

The screenshot shows the Jenkins configuration interface for the 'package_docker_v2' job. A context menu is open over the 'Post-build Actions' section, specifically over the 'Deploy war/ear to a container' option. The menu also lists other actions like 'Archive the artifacts' and 'Build other projects'. The 'Deploy war/ear to a container' option is highlighted.

Tomcat settings1

Configuration

Post-build Actions

- General
- Source Code Management
- Build Triggers
- Build Environment
- Build Steps
- Post-build Actions**

Deploy war/ear to a container

WAR/EAR files: `**/*.war`

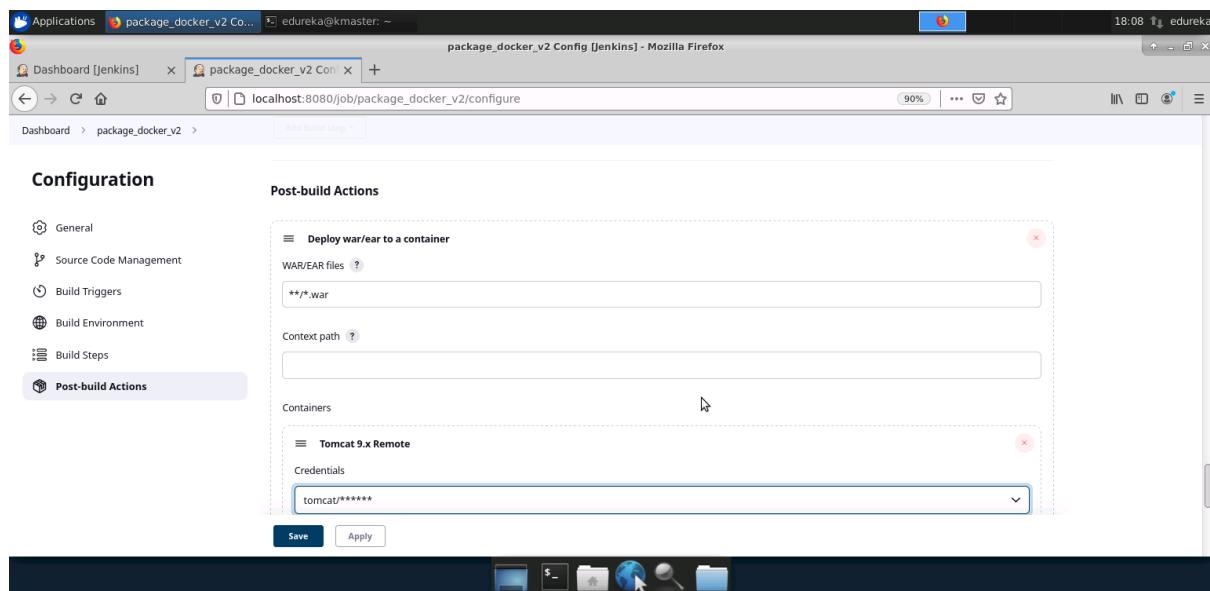
Context path: `/`

Containers

Tomcat 9.x Remote

Credentials: `tomcat/*****`

Save **Apply**



Tomcat settings2

Configuration

Post-build Actions

- General
- Source Code Management
- Build Triggers
- Build Environment
- Build Steps
- Post-build Actions**

Tomcat 9.x Remote

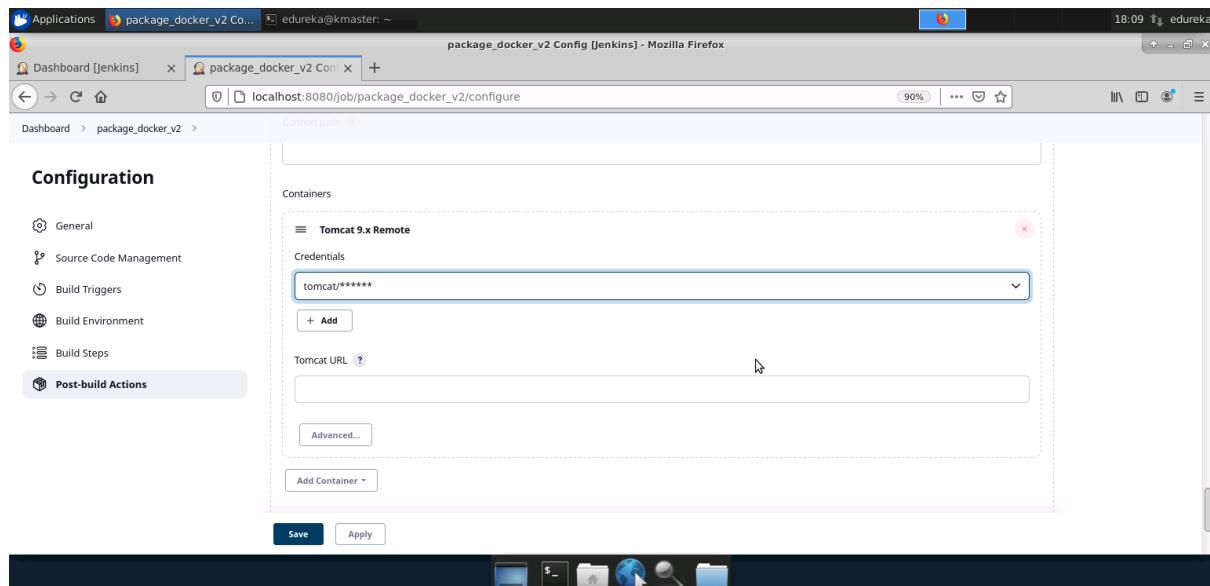
Credentials: `tomcat/*****`

Tomcat URL: `http://localhost:8080`

Advanced...

Add Container

Save **Apply**

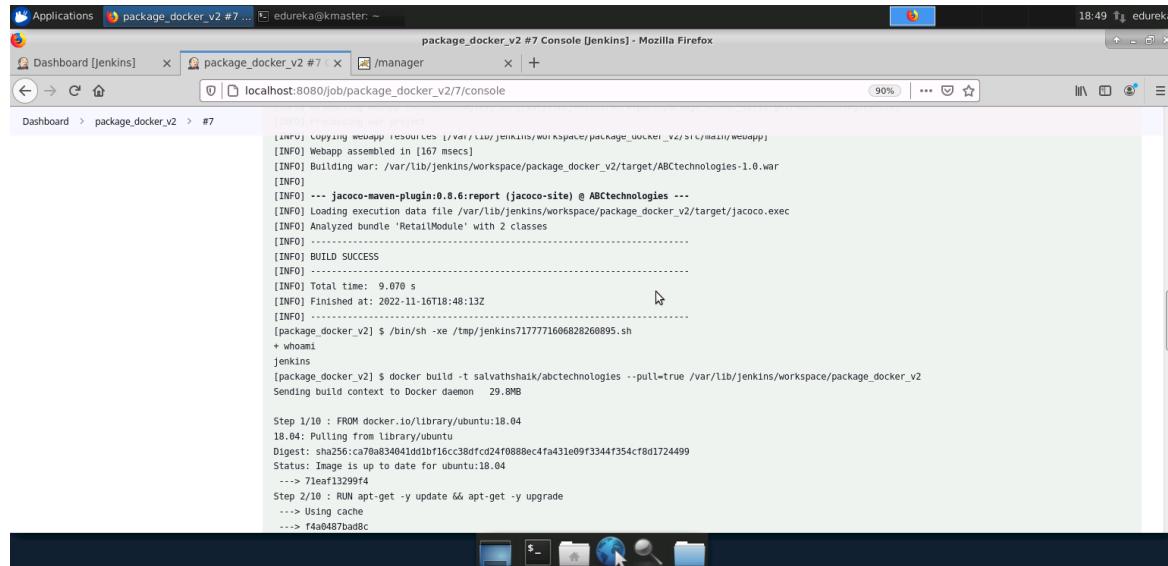


Tomcat validation1

Tomcat manager login

Tomcat manager UI that shows all the .war deployed files. We can deploy from here as well.

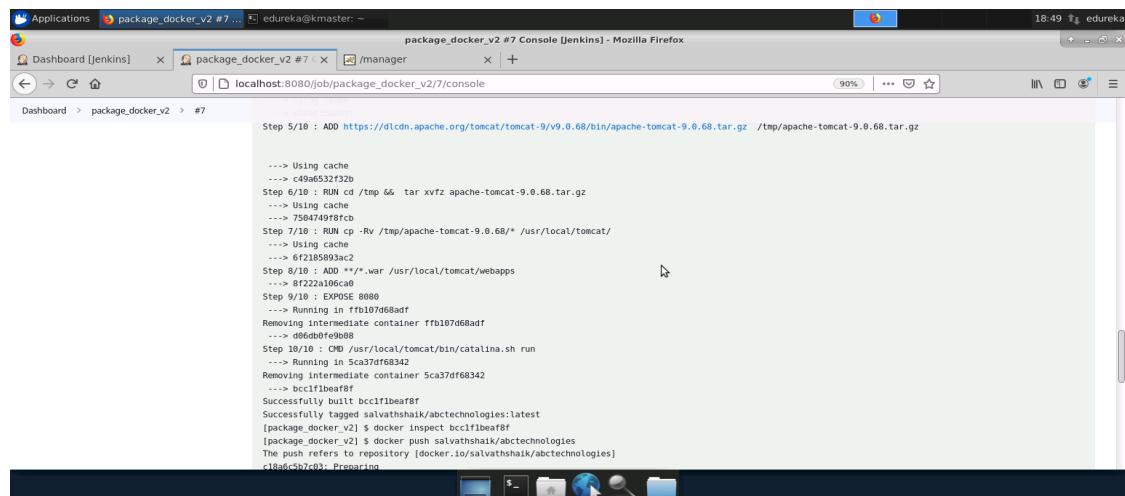
Package job success



The screenshot shows a Jenkins job named "package_docker_v2 #7" running on a Firefox browser. The console output indicates a successful build process:

```
[INFO] Copying webapp resources /var/lib/jenkins/workspace/package_docker_v2/src/main/webapp  
[INFO] Webapp assembled in [167 ms]  
[INFO] Building war: /var/lib/jenkins/workspace/package_docker_v2/target/ABCtechnologies-1.0.war  
[INFO] --- jacoco-maven-plugin:0.8.6:report (jacoco-site) @ ABCtechnologies ---  
[INFO] Loading execution data file /var/lib/jenkins/workspace/package_docker_v2/target/jacoco.exec  
[INFO] Analyzing bundle 'RetailModule' with 2 classes  
[INFO] .....  
[INFO] BUILD SUCCESS  
[INFO] .....  
[INFO] Total time: 9.070 s  
[INFO] Finished at: 2022-11-16T18:48:13Z  
[INFO] .....  
(package_docker_v2) $ /bin/sh -xe /tmp/jenkins71777160628260895.sh  
+ whoami  
jenkins  
[package_docker_v2] $ docker build -t salvathshaik/abctechnologies --pull=true /var/lib/jenkins/workspace/package_docker_v2  
Sending build context to Docker daemon 29.8MB  
Step 1/10 : FROM docker.io/library/ubuntu:18.04  
18.04: Pulling from library/ubuntu  
Digest: sha256:ca70a834041dd1bf16cc38dfcd24f0888ec4fa431e09f3344f354cf8d1724499  
Status: Image is up to date for ubuntu:18.04  
--> 7leaf13299f4  
Step 2/10 : RUN apt-get -y update && apt-get -y upgrade  
--> Using cache  
--> f4a0487bad8c
```

Docker hub push success message



The screenshot shows a Jenkins job named "package_docker_v2 #7" running on a Firefox browser. The console output shows the Docker image was successfully pushed to Docker Hub:

```
Step 5/10 : ADD https://dlcdn.apache.org/tomcat/tomcat-9/v9.0.68/bin/apache-tomcat-9.0.68.tar.gz /tmp/apache-tomcat-9.0.68.tar.gz  
--> Using cache  
--> c49eb532f23b  
Step 6/10 : RUN cd /tmp && tar xvfz apache-tomcat-9.0.68.tar.gz  
--> Using cache  
--> 7leaf13299f4  
Step 7/10 : RUN cp -Rv /tmp/apache-tomcat-9.0.68/* /usr/local/tomcat/  
--> Using cache  
--> 6f2185893ac2  
Step 8/10 : ADD **/* war /usr/local/tomcat/webapps  
--> 8f222a1a06ca0  
Step 9/10 : EXPOSE 8080  
--> Running in ffb107d68adff  
Removing intermediate container ffb107d68adff  
--> d06db0fe9b08  
Step 10/10 : CMD /usr/local/tomcat/bin/catalina.sh run  
--> Running in 5ca37df68342  
Removing intermediate container 5ca37df68342  
--> bcc11beaef8  
Successfully built bcc11beaef8  
Successfully tagged salvathshaik/abctechnologies:latest  
[package_docker_v2] $ docker inspect bcc11beaef8  
[package_docker_v2] $ docker push salvathshaik/abctechnologies  
The push refers to repository [docker.io/salvathshaik/abctechnologies]  
c18ad5b7c03: Preparing
```

Docker push and tomcat running success message

```

package_docker_v2 #7 Console [Jenkins] - Mozilla Firefox
localhost:8080/job/package_docker_v2/7/console
Dashboard > package_docker_v2 > #7

4002011234: Layer already exists
91fd0d6738b4: Layer already exists
822b35c51294: Layer already exists
69f57fbce0b1: Layer already exists
050e3c1883f2: Layer already exists
4252e6d7052: Layer already exists
c18a6c5b7c03: Pushed
latest: digest: sha256:ee7883c0004ee876d19dc1c0349831bf2f77fb1c6e8a8780bf645ff21e3dd7c size: 2008
[package_docker_v2] $ /bin/sh -xe /tmp/jenkins12858965793707520489.sh
+ echo Successfully build and uploaded to dockerhub
Successfully built and uploaded to dockerhub
+ sudo docker run -d -P salvathshaik/abctechnologies
70d295ce3f64ae4d7d0d4426a707abcc852d54b0f6ec3bd2e71d29d643e1a81a
+ echo Successfully deployed to the docker container by pulling the image from dockerhub
Successfully deployed to the docker container by pulling the image from dockerhub
[DeployPublisher][INFO] Attempting to deploy 1 war file(s)
[DeployPublisher][INFO] Deploying [/var/lib/jenkins/workspace/package_docker_v2/target/ABCtechnologies-1.0.war] to container Tomcat 9.x Remote with context null
  Redeploying [/var/lib/jenkins/workspace/package_docker_v2/target/ABCtechnologies-1.0.war]
  Undeploying [/var/lib/jenkins/workspace/package_docker_v2/target/ABCtechnologies-1.0.war]
  Deploying [/var/lib/jenkins/workspace/package_docker_v2/target/ABCtechnologies-1.0.war]
Finished: SUCCESS

```

REST API Jenkins 2.361

Validating in terminal for docker images

```

File Edit View Search Terminal Help
edureka@kmaster: ~
27 sudo chmod 777 /var/lib/jenkins/.docker/config.json
28 docker ps
29 sudo usermod -aG docker jenkins
30 newgrp - docker
31 exit
32 docker ps
33 ls -al
34 sudo chmod 777 .docker
35 docker ps
36 history
jenkins@kmaster:~$ docker images
REPOSITORY          TAG      IMAGE ID      CREATED       SIZE
salvathshaik/abctechnologies    latest      bcc1f1beaf8f  2 minutes ago  558MB
<none>              <none>      3bfb33ac76ad  17 minutes ago  558MB
<none>              <none>      e5766d5d37b9  25 minutes ago  558MB
<none>              <none>      86b69b7b34fe  26 minutes ago  558MB
<none>              <none>      53a450163d3f  27 minutes ago  558MB
<none>              <none>      fcae24101922  35 minutes ago  558MB
<none>              <none>      8b9997c49202  38 minutes ago  558MB
<none>              <none>      e0726d61ee92  9 days ago   558MB
salvathshaik/abctechnologies    <none>      a33a265f1a1   9 days ago   518MB
<none>              <none>      d66192101c64  13 days ago  59.4MB
rancher/mirrored-flannelcni-flannel  v0.20.1   a8780b506fa4  2 weeks ago  77.8MB
ubuntu                  latest      71leaf13299f4  3 weeks ago  63.1MB
rancher/mirrored-flannelcni-flannel-cni-plugin v1.1.0   fcecffc7ad4a  6 months ago  8.09MB
k8s.gcr.io/kube-proxy           v1.18.20    27ff8b8d51985  17 months ago  117MB
k8s.gcr.io/kube-apiserver        v1.18.20    7d8d2960de69  17 months ago  173MB
k8s.gcr.io/kube-scheduler        v1.18.20    a05a1a79adaa  17 months ago  96.1MB
k8s.gcr.io/kube-controller-manager v1.18.20    e7c545a60706  17 months ago  162MB
k8s.gcr.io/pause                 3.2        80d28bedfe5d  2 years ago   683kB
k8s.gcr.io/coredns               1.6.7      67da37a9a360  2 years ago   43.8MB
k8s.gcr.io/etcd                  3.4.3-0    303ce5db0e90  3 years ago   288MB
jenkins@kmaster:~

```

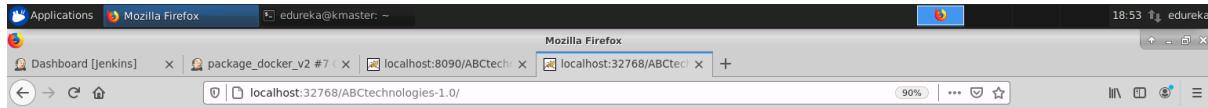
Validating container is running or not

```

File Edit View Search Terminal Help
k8s.gcr.io/kube-scheduler v1.18.20 a05ala79adaa 17 months ago 96.1MB
k8s.gcr.io/kube-controller-manager v1.18.20 e7c545a60706 17 months ago 162MB
k8s.gcr.io/pause 3.2 88d28bedfe5d 2 years ago 683kB
k8s.gcr.io/coredns 1.6.7 67da37a9a360 2 years ago 43.8MB
k8s.gcr.io/etcd 3.4.3-0 303ce5db0e90 3 years ago 288MB
jenkins@kmaster:~$ docker ps
CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES
70d295ce3f64 salvathshaik/abctechnologies "/bin/sh -c '/usr/lo..." 2 minutes ago Up 2 minutes 0.0.0.0:32768->8080/tcp k8s_quizzical_raman
f5f4090d6b7b d66192101c64 "/opt/bin/flanneld -..." 13 minutes ago Up 13 minutes k8s_kube-flannel_kube-flan
nel-ds-xr4m7_kube-flannel_4fd68a30-73f7-436b-85a5-291aca80d63bb_20
0e45920f908a 70d8d2960de9 "kube-apiserver --ad..." 13 minutes ago Up 13 minutes k8s_kube-apiserver_kube-ap
iserver-kmaster_kmaster_kube-system_6b062f177409020443c97cd3a3009ee6b_20
c800fa81ceca3 67da37a9a360 "/coredns -conf /etc..." 13 minutes ago Up 13 minutes k8s_coredns_coredns-66bff4
67f8b4sxtk_kube-system_3858ccb7-7461-4e95-84db-37619107233f_19
ddf357b7d79c a05ala79adaa "kube-scheduler --au..." 13 minutes ago Up 13 minutes k8s_kube-scheduler_kube-sc
heduler-kmaster_kube-system_e541886a4cda0424b9879a78869ad51_20
2ac7f91f3dfc 27fb8bd51985 "/usr/local/bin/kube..." 13 minutes ago Up 13 minutes k8s_kube-proxy_kube-proxy-
rbfw8_kube-system_bbd75397-5d66-4916-bfb7-7344a8292b06_19
42dd9e3375f7 e7c545a60706 "kube-controller-man..." 13 minutes ago Up 13 minutes k8s_kube-controller-manage
r_kube-controller-manager-kmaster_kube-system_2cf16e7db66397e307ead965fc8a447_19
c4988cf5596f 303ce5db0e90 "/etc --advertise-cl..." 13 minutes ago Up 13 minutes k8s_etcd_etcd-kmaster_kube
-system_a4e1cedd69431ed65bf4d3c49f6111c5_19
3452e2302fa3 k8s.gcr.io/pause:3.2 "/pause" 14 minutes ago Up 13 minutes k8s_POD_kube-apiserver-kma
ster_kube-system_6b062f177409020443c97cd3a3009ee6b_20
53347eacf205 k8s.gcr.io/pause:3.2 "/pause" 14 minutes ago Up 13 minutes k8s_POD_coredns-66bff467f8
-4sxtk_kube-system_3858ccb7-7461-4e95-84db-37619107233f_76
a3402d1f0f9f k8s.gcr.io/pause:3.2 "/pause" 14 minutes ago Up 13 minutes k8s_POD_kube-flannel-ds-xr
4m7_kube-flannel_4fd68a30-73f7-436b-85a5-291aca80d63bb_20
3a6a294acf47 k8s.gcr.io/pause:3.2 "/pause" 14 minutes ago Up 13 minutes k8s_POD_kube-controller-ma
nager-kmaster_kube-system_2cf16e7db66397e307ead965fc8a447_19
07b1699cf1b k8s.gcr.io/pause:3.2 "/pause" 14 minutes ago Up 13 minutes k8s_POD_etcd-kmaster_kube-
system_a4e1cedd69431ed65bf4d3c49f6111c5_19
6e168847cad5 k8s.gcr.io/pause:3.2 "/pause" 14 minutes ago Up 13 minutes k8s_POD_kube-proxy-rbfw8_k
ube-system bbd75397-5d66-4916-bfb7-7344a8292b06_19

```

Docker container validation in browser



Welcome to ABC technologies

This is retail portal

[Add Product](#) [View Product](#)



Tomcat browser validation

Welcome to ABC technologies

This is retail portal

Add Product View Product

View name

CL_CD_DOCKER_V2

Type

Build Pipeline View

Include a global view

List View

My View

create

Based on upstream/downstream relationship

Upstream / downstream config

Select Initial Job

compile

Trigger Options

Build Cards

Standard build card

Use the default build cards

Restrict triggers to most recent successful builds

Yes

OK Apply

Modifying the job configuration for pipeline connectivity

The screenshot shows the Jenkins job configuration page for 'package_docker_v2'. The 'Build Triggers' section is active. Under 'Build after other projects are built', the checkbox is checked. In the 'Projects to watch' input field, 'test' is typed. Below this, there are several trigger options: 'Trigger only if build is stable' (radio button selected), 'Trigger even if the build is unstable', 'Trigger even if the build fails', and 'Always trigger, even if the build is aborted'. There are also sections for 'Build periodically', 'GitHub hook trigger for GITScm polling', and 'Poll SCM'. At the bottom are 'Save' and 'Apply' buttons.

Before triggering pipeline view

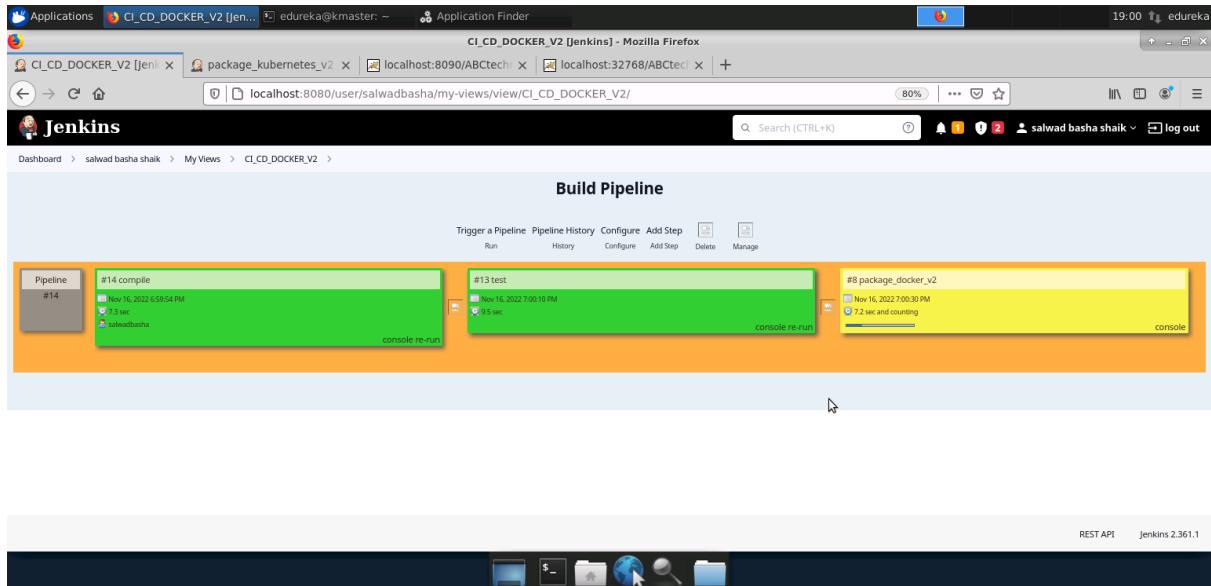
The screenshot shows the Jenkins 'Build Pipeline' view for the 'CI_CD_DOCKER_V2' project. The pipeline consists of three stages: '#13 compile', 'test', and 'package_docker_v2'. The 'test' stage is currently active, indicated by a yellow background. The pipeline has been triggered and is running, as shown by the progress bar and timestamp ('Nov 16, 2022 4:19:12 PM'). The Jenkins interface includes a top navigation bar with tabs like 'Trigger a Pipeline', 'Pipeline History', 'Configure', 'Add Step', 'Run', 'History', 'Configure', 'Add Step', 'Delete', and 'Manage'.

Started pipeline

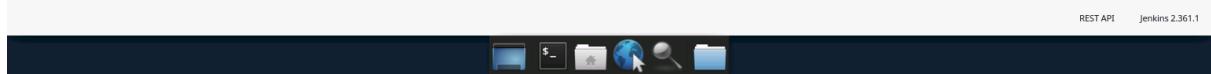
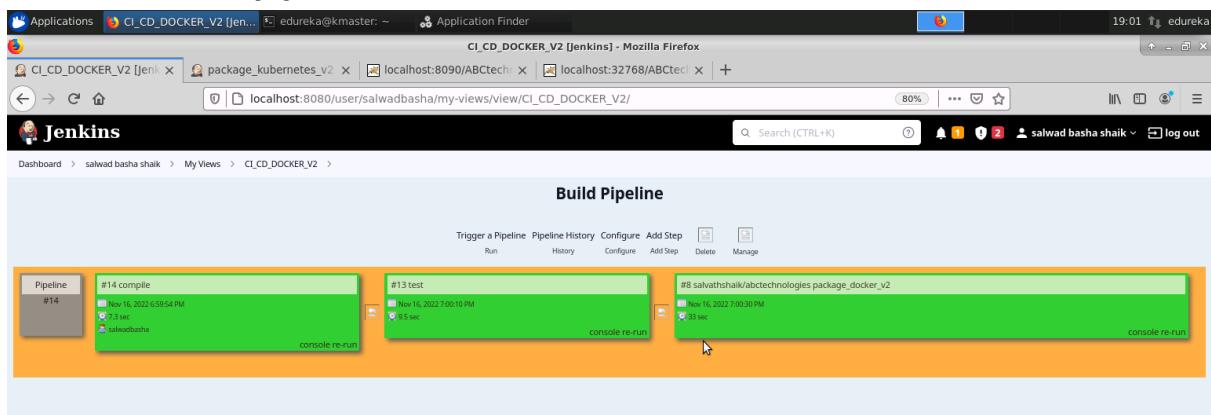
The screenshot shows the Jenkins 'Build Pipeline' view for the 'CI_CD_DOCKER_V2' project. The pipeline has started, and the first stage, '#14 compile', is now highlighted in yellow, indicating it is currently running. The timestamp shows 'Nov 16, 2022 6:59:54 PM'. The other stages, 'test' and 'package_docker_v2', are shown in blue and have 'N/A' status. The Jenkins interface includes a top navigation bar with tabs like 'Trigger a Pipeline', 'Pipeline History', 'Configure', 'Add Step', 'Run', 'History', 'Configure', 'Add Step', 'Delete', and 'Manage'.



Last job running in pipeline



Success run for pipeline



Validating in build and showing Test job as a downstream for compile job

The screenshot shows the Jenkins 'compile' project page. On the left, there's a sidebar with links like 'Back to Dashboard', 'Status', 'Changes', 'Workspace', 'Build Now', 'Configure', 'Delete Project', 'Git Polling Log', and 'Rename'. The main content area has a heading 'Project compile'. Below it, there's a 'Status' section with a note: 'this is for compiling the java source code.' A 'Downstream Projects' section shows a blue link labeled 'test'. A 'Permalinks' section contains a link to 'test'. A 'Build History' section shows two builds: #14 (Nov 16, 2022, 6:59 PM) and #13 (Nov 16, 2022, 4:19 PM). At the top right, there are 'Edit description' and 'Disable Project' buttons.

Test results upstream details

The screenshot shows the Jenkins 'test' project page. The sidebar is identical to the 'compile' project page. The main content area has a heading 'Project test'. It shows a 'Latest Test Result (no failures)' icon and a 'Upstream Projects' section with a blue link labeled 'compile'. A 'Downstream Projects' section shows a blue link labeled 'package_docker_v2'. A 'Build History' section shows four builds: #13 (Nov 16, 2022, 7:00 PM), #11 (Nov 9, 2022, 4:34 PM), #10 (Nov 7, 2022, 7:33 PM), and #9 (Nov 4, 2022, 7:00 AM). To the right, there's a 'Test Result Trend' chart with a green shaded area representing 'Passed' status across eleven build numbers. At the top right, there are 'Add description' and 'Disable Project' buttons.

Validating who is started the test job

The screenshot shows the Jenkins interface for a job named 'test' (Build #13). The main title is 'Build #13 (Nov 16, 2022, 7:00:10 PM)'. The status is green with a checkmark. On the left, there's a sidebar with links like 'Back to Project', 'Status' (which is selected), 'Changes', 'Console Output', 'Edit Build Information', 'Delete build #13', 'Git Build Data', 'Test Result', and 'Previous Build'. The 'Changes' section lists 11 commits from upstream project 'compile' build number 14. Below it, a 'git' badge indicates the revision and repository. The right side shows deployment details: 'Started by user salwad basha shaik' and 'Started 3 min 20 sec ago Took 9.5 sec on built-in'. A 'Keep this build forever' button is also present.

Success log of pipeline last job(package_docker_v2)

The screenshot shows the Jenkins Pipeline console output for a job named 'package_docker_v2' (Run #8). The title is 'package_docker_v2 #8 salvathshalk/abctechnologies Console [Jenkins]'. The log output shows the process of building a Docker image and pushing it to DockerHub, followed by deploying it to a Tomcat container. Key logs include:

```
42525eb7b522: Waiting
69f5f7bce01b: Waiting
48d02fbfa1934: Layer already exists
821b95c512094: Layer already exists
911d096730b4: Layer already exists
717f499ee2f2: Layer already exists
69f5f7bce01b: Layer already exists
89e03c1883f2: Layer already exists
42525eb7b522: Layer already exists
20e19d3c2e3d: Pushed
latest: digest: sha256:2ff94efad15eeefbc2bd66e8080dcc6bed016496e84fa04403621feb80c1af013 size: 2008
[package_docker_v2] $ /bin/sh -xe /tmp/jenkins1231054676731975612.sh
+ echo Successfully build and uploaded to dockerhub
Successfully build and uploaded to dockerhub
+ sudo docker run -d -P salvathshalk/abctechnologies
187bb6a67550f0c8968e0876ed9d979070191699953e49d3795094cf35f56ce1
+ echo Successfully deployed to the docker container by pulling the image from dockerhub
Successfully deployed to the docker container by pulling the image from dockerhub
[DeployPublisher][INFO] Attempting to deploy 1 war file(s)
[DeployPublisher][INFO] Deploying /var/lib/jenkins/workspace/package_docker_v2/target/ABCtechnologies-1.0.war to container Tomcat 9.x Remote with context null
Redeploying /var/lib/jenkins/workspace/package_docker_v2/target/ABCtechnologies-1.0.war
Undeploying /var/lib/jenkins/workspace/package_docker_v2/target/ABCtechnologies-1.0.war
Deploying /var/lib/jenkins/workspace/package_docker_v2/target/ABCtechnologies-1.0.war
Finished: SUCCESS
```

The bottom of the screen shows Jenkins navigation icons and the text 'REST API Jenkins 2.361.1'.

Docker-hub validation after pushing from jenkins:

The screenshot shows the Docker Hub interface for the repository 'abctechnologies'. The 'Tags' tab is selected, displaying one tag: 'v1'. The tag was last pushed 31 minutes ago by the user 'salvathshaik'. The image details include the digest '915822897a73', OS/ARCH 'linux/amd64', and a compressed size of 230.16 MB.

References i have followed:

- <https://plugins.jenkins.io/role-strategy/>
- <https://stackoverflow.com/questions/37603621/jenkins-sudo-no-tty-present-and-no-askpass-program-specified-with-nopasswd>
- <https://dlcdn.apache.org/tomcat/tomcat-9/v9.0.68/bin/apache-tomcat-9.0.68.tar.gz>
- <https://phoenixnap.com/kb/how-to-configure-docker-in-jenkins>
- https://www.tutorialspoint.com/docker/docker_continuous_integration.htm
- <https://www.provartesting.com/documentation/devops/continuous-integration/docker/setting-up-continuous-integration-with-jenkins-for-docker/>
- <https://www.youtube.com/watch?v=mszE-OCI2V4&list=PLVz2XdJiJQxwS0BZUHX34ocLTJtRGSQzN&index=4>
- <https://www.digitalocean.com/community/questions/how-to-fix-docker-got-permission-denied-whiletryingtoconnecttothedockerdaemonsocket>
- <https://stackoverflow.com/questions/50798720/jenkins-throwing-error-jenkins-model-invalidbuildsdir-item-rootdir-builds-d>
- <https://linuxize.com/post/how-to-list-groups-in-linux/>
- <https://stackoverflow.com/questions/17733671/how-can-i-tell-what-user-jenkins-is-running-as>
- <https://stackoverflow.com/questions/55156958/jenkins-fail-to-deploy-war-to-tomcat-container-second-time>

Task 4: Integrate the Docker host with Ansible. Write an Ansible playbook to create an image and create a continuer. Integrate Ansible with Jenkins. Deploy Ansible-playbook. CI/CD job to build code on ansible and deploy it on docker container.

1. Deploy Artifacts on Kubernetes
2. Write pod, service, and deployment manifest file
3. Integrate Kubernetes with Ansible
4. Ansible playbook to create deployment and service

Approach i have followed:

- The task-4 starting of integrating the jenkins with ansible. and creating the ansible playbook integration of docker and then kubernetes.
- And this task is an extension of the last task so I have used the DockerFile I prepared in the last task and added a few more steps.
- First of all i have configured the ansible hosts file with localhost and copied the self public key to the master machine as jenkins needs authentication internally to connect to the localhost(self) otherwise i will get authentication errors.

edureka lab has installed the required tools jenkins,ansible,docker,kubernetes

Kubernetes version:

Client Version: v1.18.3
Server Version: v1.18.20

Ansible version:

```
ansible 2.9.17
config file = /etc/ansible/ansible.cfg
configured module search path = [u'/home/edureka/.ansible/plugins/modules',
u'/usr/share/ansible/plugins/modules']
ansible python module location = /usr/lib/python2.7/dist-packages/ansible
executable location = /usr/bin/ansible
python version = 2.7.17 (default, Jul 20 2020, 15:37:01) [GCC 7.5.0]
```

So i have installed the ansible plugin in jenkins to run the playbook with provided ansible-playbook and inventory file paths to run the ansible-playbook.

The ansible playbook created with docker module to build the image and then start the build container with provided .war file path in Dockerfile. and i have used two docker modules 'community.docker.docker_image' and 'community.docker.docker_container' to build the image and run the container.

the **ansible-playbook path**: project_required_file_v2/ansible.yml in github
repo=<https://github.com/salvathshaik/final-devops-project.git>

inventory file path: project_required_file_v2/hosts

And installed the '**docker**' python module to run the docker using ansible playbook and '**docker**' module should be run for python **version>=2.6**

the edureka master machine has python version 3.8. So I have installed this 'docker' module. and modified the ansible inventory file(hosts) in such a way that my ansible should run using python3.8 version not as a default python2 version. i have modified this inventory with '**ansible_python_interpreter="/usr/bin/python3.8"**' parameter along with hostname otherwise it will use the default python2 version and it will not run using the 'docker' module.

To install this module: pip install docker

If the edureka lab has python version 2.6 i will have to install the 'docker-py' module and dont need to change the ansible inventory file as ansible use python default version python2.

To install this module: pip install docker-py

and i have installed the ansible module 'community.docker' to build the docker image and run the container. and this has to be installed in jenkins user separately otherwise it won't recognize the module and will throw the error so this is very important that i need to install this module for specific the user that i am running the ansible playbook.

To install this module : ansible-galaxy collection install community.docker

Jenkins and ansible integration process to build the docker image, run the build image container and then deploy to kubernetes with deployment,service(nodeport).

Note: I have made sure to copy the deployment.yml and Dockerfile file in the '/home/edureka/Desktop' folder before doing the below setup.

First i have installed the ansible plugin in manage plugins section of jenkins-> and created the job 'package_kubernetes_v2' and configured this by giving the git path with main branch-> and selected the 'Execute shell' in build step for login to the docker with credentials-> and then choose the 'Invoke Ansible Playbook' and provided the given playbook path 'project_required_file_v2/ansible.yml', inventory file path 'project_required_file_v2/hosts'. -> and then run the build.

And then created the CI/CD pipeline view as '**CI_CD_KUBERNETES_V2**' and then ran the pipeline.

command for docker login: sudo docker login -u salvathshaik -p <password>

```

ansible file:
project_required_file_v2/ansible.yml
---
- hosts: localhost
  become: yes
  tasks:
    - name: Build an image and push it to a private repo
      community.docker.docker_image:
        build:
          path: "/home/edureka/Desktop"
          name: docker.io/salvathshaik/abctechnologies
          #tag: test
          push: true
          source: build
        register: out
    - debug:
        var: out
    - name: start a container
      community.docker.docker_container:
        name: abc-application
        image: salvathshaik/abctechnologies
        state: started
        ports:
          - 1234:8080
        #restart: true
      register: out
    - debug:
        var: out
      #- name: Deploying to kubernetes
    - name: Create a Deployment by reading the definition from a local file
      command: "kubectl --kubeconfig=/etc/kubernetes/admin.conf apply -f
/home/edureka/Desktop/deployment.yml"
      register: out
    - debug:
        var: out
---

```

There are 3 tasks created above in an ansible file.

- 1) Building the docker image and pushing to docker hub(- name: Build an image and push it to a private repo)
- 2) Run the build docker container (- name: start a container)
- 3) deploying artifacts to kubernetes with deployment controller and NodePort service (Create a Deployment by reading the definition from a local file)

- **In first task**, i am taking a Dockerfile from path "/home/edureka/Desktop" to build and pushing to docker hub repository (docker.io/salvathshaik/abctechnologies) and displaying the output using '**-debug module**' in ansible.

- The above code will start to build the docker image by taking the Dockerfile from '/home/edureka/Desktop' directory.

- **In second task**, i am pulling the docker image (salvathshaik/abctechnologies) and running with custom port exposed to 1234 by providing container port as 8080 and displaying the output using '**-debug**' module in ansible.

- **In third task**, using ansible 'command' module to be used as a shell to run the kubernetes command to run the deployment.yml file.

And I have given the config path in the ansible file and provided the **become: sudo** access to use the admin.conf file to run the deployment.

As part of readiness i have tried to run the ansible-playbook locally not by jenkins and then tried in jenkins.

Process of deploying the artifacts to kubernetes using ansible i was seen one k8s kubernetes module available in openshift but i observed in-compatibility issues with the python. Ansible using python version 2.7.17 internally after installation of ansible but for k8s module of openshift requires more than 3.6 version even after i followed many solutions online still ansible could not taking python version as 3.8 or 3.6 and even after i modified some configurations in the ansible.conf file it's not loading the k8s module from openshift unfortunately this all took more than 8 hours for me but still not worked out So i have followed the approach to run the ansible play-book from jenkins and internally jenkins will integrate ansible to run the playbook to build the docker image and run the docker container and then created the deployment and service(NodePort) to deploy our latest version of an application.

register: out

 - **debug:**

var: out

explanation: Here we need to put the register keyword just below the output you want to store.

For ex: I have put the 'register' module just below the 'command' module so after executing this 'command' module line it will store the output (error or success) to 'out' variable only, not any other variable.

And in debug we need to make sure to put 'var' if we are displaying any output of some command like above. And 'msg' is like echo in linux.

inventory file**project_required_file_v2/hosts**

```
[local]
```

```
localhost ansible_python_interpreter="/usr/bin/python3.8"
```

Dockerfile

```
/home/edureka/Desktop/Dockerfile
FROM docker.io/library/ubuntu:18.04
RUN apt-get -y update && apt-get -y upgrade
RUN apt-get -y install openjdk-8-jdk wget
RUN mkdir /usr/local/tomcat
ADD
https://dlcdn.apache.org/tomcat/tomcat-9/v9.0.68/bin/apache-tomcat-9.0.68.tar.gz
/tmp/apache-tomcat-9.0.68.tar.gz
RUN cd /tmp && tar xvfz apache-tomcat-9.0.68.tar.gz
RUN cp -Rv /tmp/apache-tomcat-9.0.68/* /usr/local/tomcat/
ADD **/*.war /usr/local/tomcat/webapps
EXPOSE 8080
CMD /usr/local/tomcat/bin/catalina.sh run
```

explanation: taking an Ubuntu image and modifying the image with installation of java and tomcat and then deploying the .war file to the tomcat container and running this container.

And Prepared the Deployment and service files for deploying the application.

```
deployment.yml
/home/edureka/Desktop/deployment.yml
---
kind: Deployment
apiVersion: apps/v1
metadata:
  name: abctechnologies-dep
spec:
  replicas: 2
  minReadySeconds: 45
  strategy:
    type: RollingUpdate
    rollingUpdate:
      maxUnavailable: 1
      maxSurge: 2
  selector:
    matchLabels:
      app: abc-tech-app
  template:
    metadata:
      labels:
        app: abc-tech-app
    spec:
      containers:
        - image: salvathshaik/abctechnologies
          name: app
---

```

```
kind: Service
apiVersion: v1
metadata:
  name: abc-tech-service
spec:
  type: NodePort
  selector:
    app: abc-tech-app
  ports:
    - port: 80 #cluster port
      targetPort: 8080 #container image port

```

explanation:

- acquiring 2 replica pods that match with label 'abc-tech-app' if not found then creating pods using template using docker image repository (salvathshaik/abctechnologies) and exposing the created pods using service type 'NodePort' for accessing outside of network.

- The deployment file includes service also and has configured the strategy as below in deployment. The strategy is this deployment needs to follow rolling update strategy which will upgrade the container with requested version one by one and destroy the older version one by one.

minReadySeconds: 45 : indicates to wait for 45 seconds time to prepare the new version pods to initialize (based on maxSurge value)

and this maxUnavailable: 1: indicates that the requested replicas count for the new version should not be less than 1 while making up the new version pods for application availability.

Pod also created but as i have created deployment already with 2 pod replicas so i did not used it

```
kind: Pod
apiVersion: v1
metadata:
  name: abctechnologies-pod
  namespace: default
  labels:
    role: abc-tech-app
spec:
  containers:
    - image: salvathshaik/abctechnologies:docker_tag
      name: app
```

While doing this task i have encounter some issues as below

Issue1:

Permission error

Solution:

And add this permission for jenkins user in /etc/sudoers file,
jenkins ALL=(ALL) NOPASSWD: ALL

Issue2;

Failed to connect to the host via ssh: Warning: Permanently added 'localhost' (ECDSA) to the list of known hosts.

jenkins@localhost: Permission denied (publickey,password).

Solution:

So added the ssh key(public) to the self jenkins user as well then it works without giving the above error.

Please find the screenshots of all steps that i have followed:

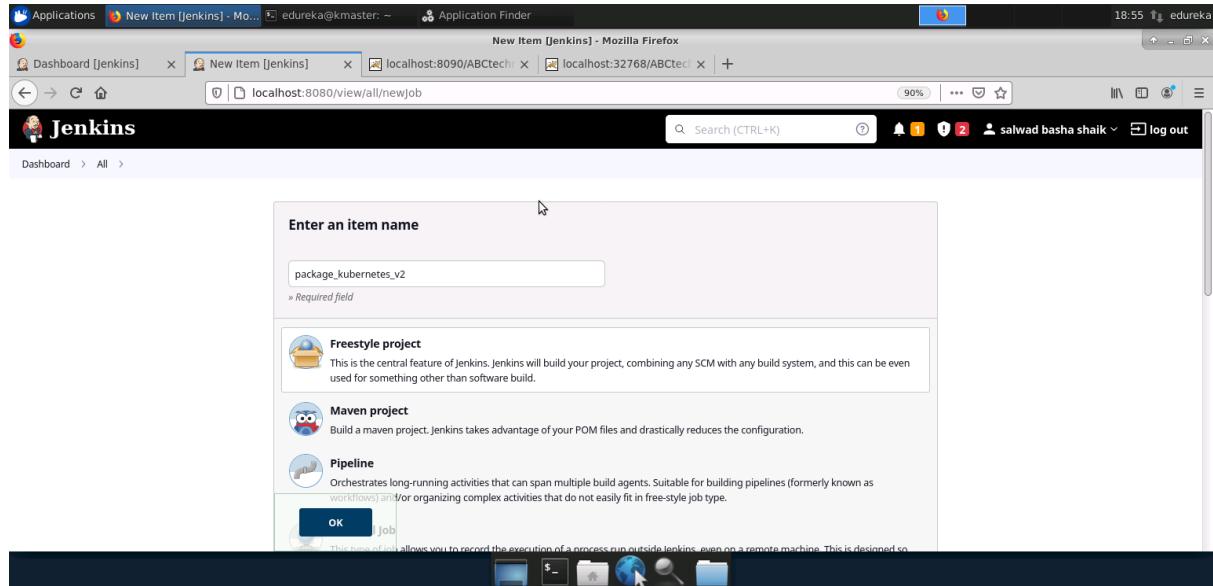
Downloading the ansible plugin

The screenshot shows the Jenkins Plugin Manager interface. The search bar at the top contains the text 'ansible'. Below the search bar, there is a table with two rows of plugin information. The first row is for 'Ansible 1.1', which is checked and has a 'Released' status with a timestamp of '2 yr 0 mo ago'. The second row is for 'Ansible Tower 0.16.0', which is unchecked and has a 'Released' status with a timestamp of '2 yr 5 mo ago'. At the bottom of the page, there are three buttons: 'Install without restart', 'Download now and install after restart', and 'Check now'. The status bar at the bottom right indicates 'REST API Jenkins 2.361.1'.

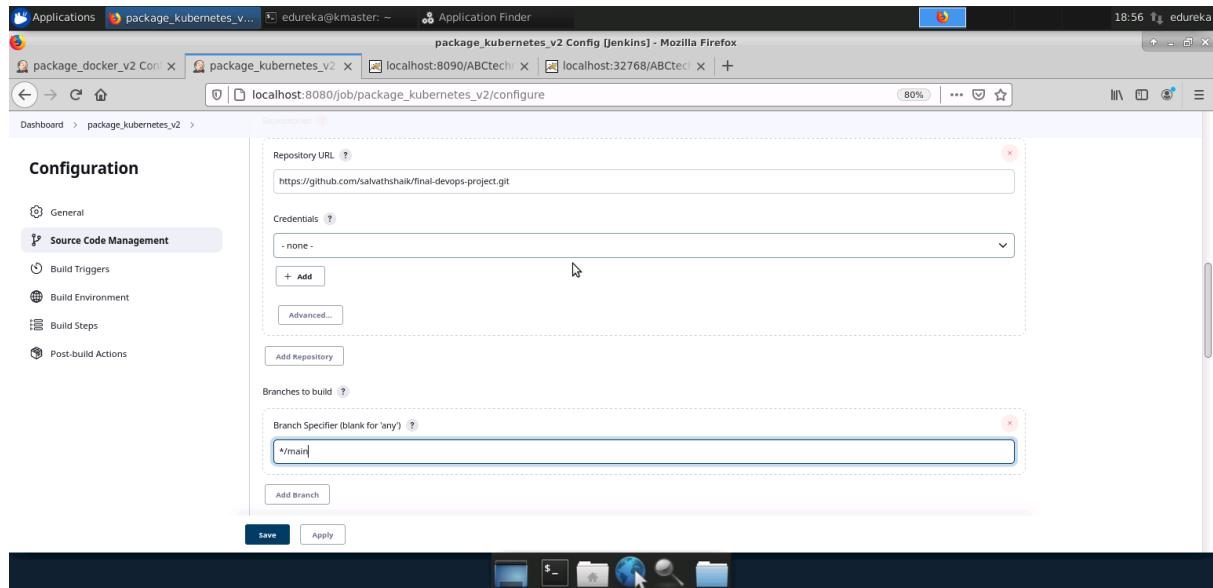
Plugin download successful screen

The screenshot shows the Jenkins Update Center interface. The main title is 'Installing Plugins/Upgrades'. On the left, there is a sidebar with links for 'Back to Dashboard', 'Manage Jenkins', and 'Manage Plugins'. The main content area shows a 'Preparation' section with a bulleted list: '• Checking internet connectivity', '• Checking update center connectivity', and '• Success'. Below this, there is a section for 'Ansible' with the message 'Loading plugin extensions' and two green checkmark icons labeled 'Success'. At the bottom, there are two buttons: 'Go back to the top page' and 'Restart Jenkins when installation is complete and no jobs are running'. The status bar at the bottom right indicates 'REST API Jenkins 2.361.1'.

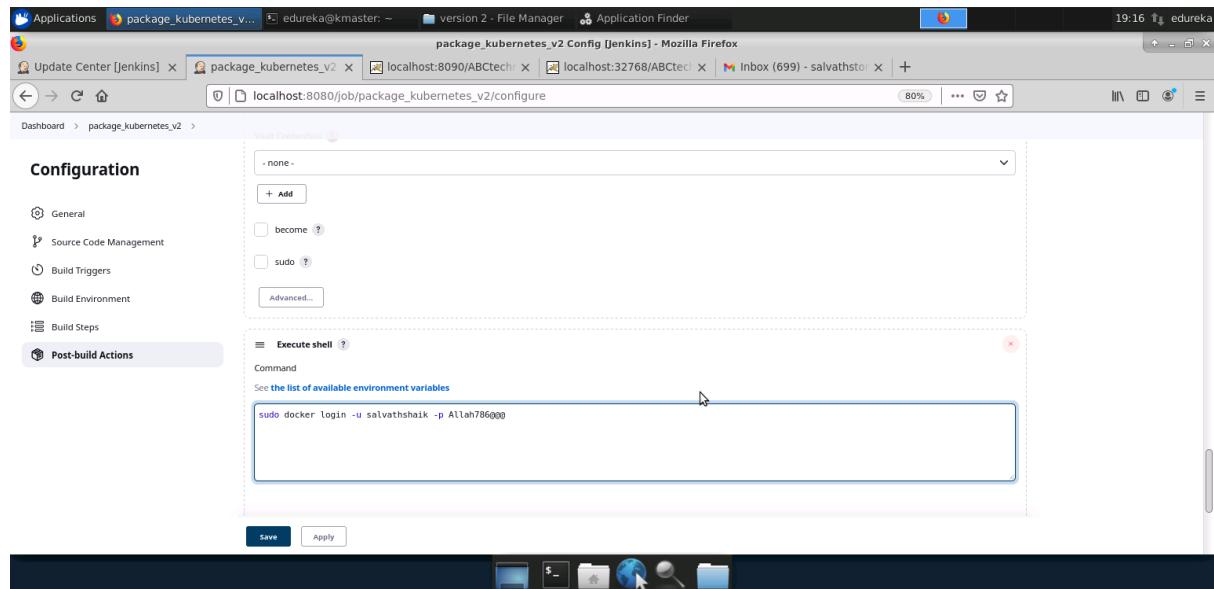
Creating the job(package_kubernetes_v2)



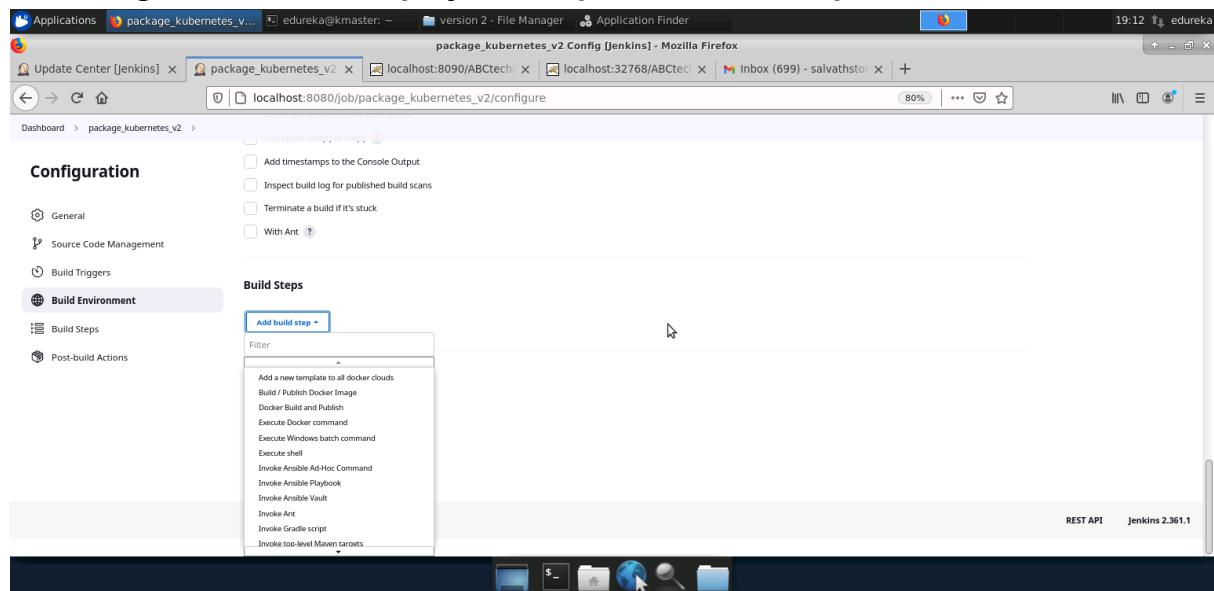
Setting up the git repository1



Build step 'Execute shell' to login to the docker hub



Choosing 'Invoke ansible playbook' option in build step



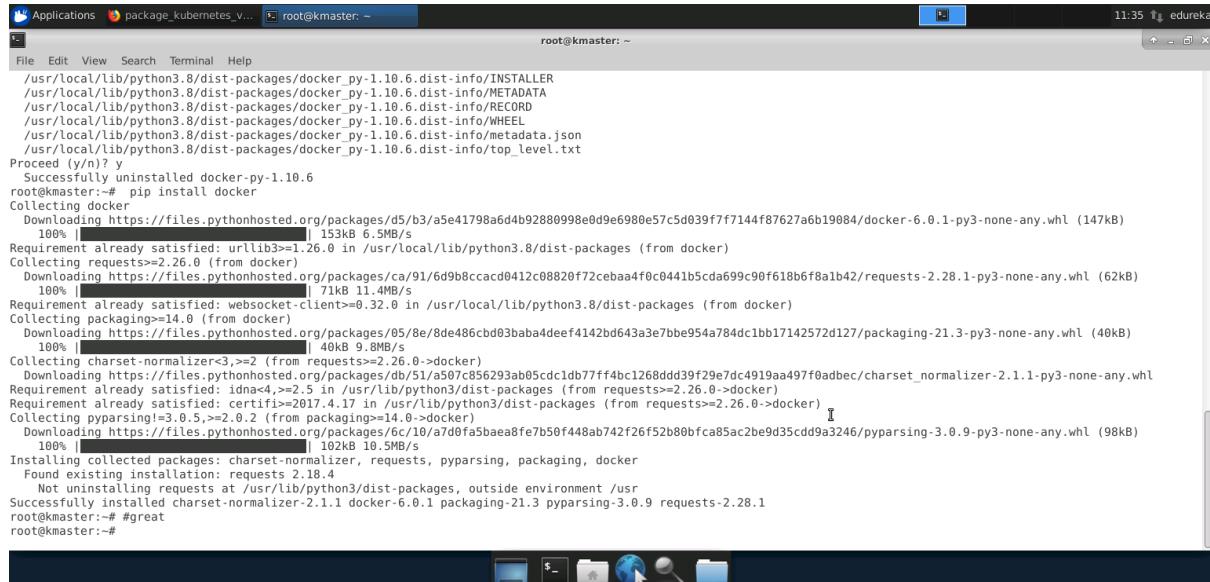
Giving ansible-playbook path to run from jenkins

The screenshot shows the Jenkins configuration interface for a job named 'package_kubernetes_v2'. The 'Build Steps' section is selected. Under 'Invoke Ansible Playbook', the 'Playbook path' field contains 'project_required_file_v2/ansible.yml'. The 'Inventory' section has 'Do not specify Inventory' selected. The 'Host subset' and 'Credentials' fields are empty. At the bottom are 'Save' and 'Apply' buttons.

Giving ansible inventory file(hosts) to use it to run the playbook

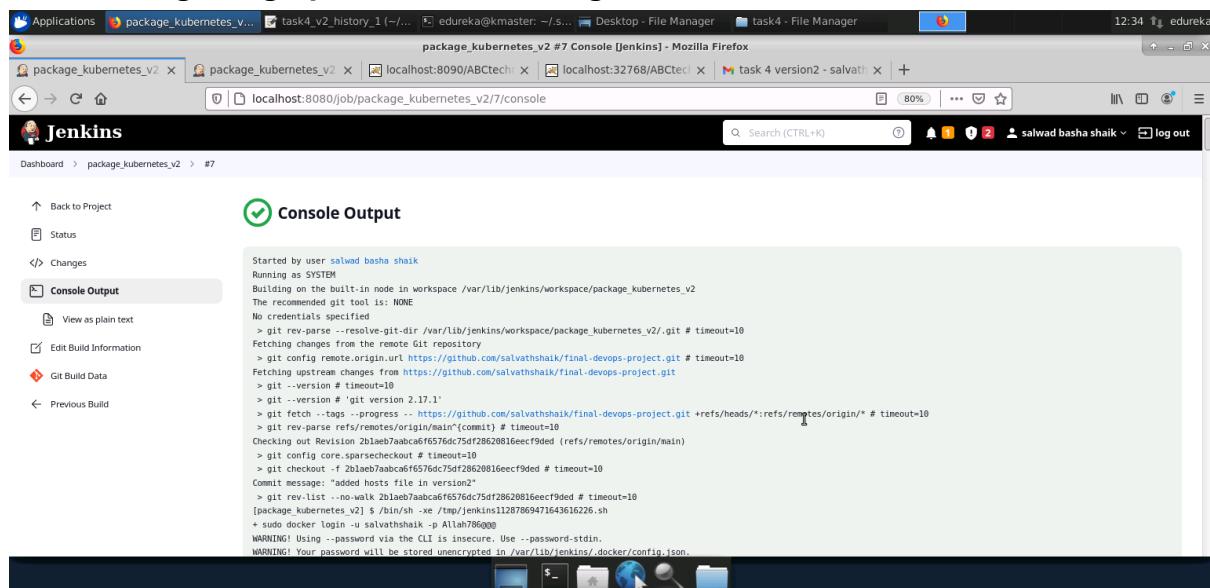
The screenshot shows the Jenkins configuration interface for a job named 'package_kubernetes_v2'. The 'Build Steps' section is selected. Under 'File path or comma separated host list', the field contains 'project_required_file_v2/hosts'. The 'Inventory' section has 'Inline content' selected. The 'Host subset' and 'Credentials' fields are empty. At the bottom are 'Save' and 'Apply' buttons.

Install the python 'docker' module as my python version is >=2.6



```
root@kmaster:~# pip install docker
Collecting docker
  Downloading https://files.pythonhosted.org/packages/d5/a5e41798a6d4b92880998e0d9e6980e57c5d039f7f144f87627a6b19084/docker-6.0.1-py3-none-any.whl (147kB)
    100% |██████████| 153kB 6.5MB/s
Requirement already satisfied: urllib3>=1.26.0 in /usr/local/lib/python3.8/dist-packages (from docker)
Collecting requests==2.26.0 (from docker)
  Downloading https://files.pythonhosted.org/packages/ca/91/6d9b8ccacd0412c08820f72cebab4f0c0441b5cda699c90f618b6fb8alb42/requests-2.28.1-py3-none-any.whl (62kB)
    100% |██████████| 71kB 11.4MB/s
Requirement already satisfied: websocket-client==0.32.0 in /usr/local/lib/python3.8/dist-packages (from docker)
Collecting packaging==14.0 (from docker)
  Downloading https://files.pythonhosted.org/packages/05/8e/8de486cbd03babaf4def4142bd643a3e7bbe954a784dc1bb17142572d127/packaging-21.3-py3-none-any.whl (40kB)
    100% |██████████| 40kB 9.8MB/s
Collecting charset-normalizer<3,>=2 (from requests>=2.26.0->docker)
  Downloading https://files.pythonhosted.org/packages/db/51/a507c507d6293ab05cc1db77ff4bc1268ddd39f29e7dc4919aa497f0adbec/charset_normalizer-2.1.1-py3-none-any.whl
Requirement already satisfied: idna<4,>=2.5 in /usr/lib/python3/dist-packages (from requests>=2.26.0->docker)
Requirement already satisfied: certifi>=2017.4.17 in /usr/lib/python3/dist-packages (from requests>=2.26.0->docker)
Collecting pyparsing!=3.0.5,>=2.0.2 (from packaging==14.0->docker)
  Downloading https://files.pythonhosted.org/packages/6c/10/a7d0fa5baea8fe7b50f448ab742f26f52b80fcac85ac2be9d35cd9a3246/pyparsing-3.0.9-py3-none-any.whl (98kB)
    100% |██████████| 102kB 10.5MB/s
Installing collected packages: charset-normalizer, requests, pyparsing, packaging, docker
  Found existing installation: requests 2.18.4
  Not uninstalling requests at /usr/lib/python3/dist-packages, outside environment /usr
Successully installed charset-normalizer-2.1.1 docker-6.0.1 packaging-21.3 pyparsing-3.0.9 requests-2.28.1
root@kmaster:~# #great
root@kmaster:~#
```

After running the build after setting up it got success
Git started getting updated code from github



The Jenkins console output shows a successful build for the 'package_kubernetes_v2' project. The build was triggered by a GitHub push event. The output details the steps taken by Jenkins, including cloning the repository, fetching upstream changes, and checking out the latest commit. It also shows the configuration of the git tool and the execution of Docker commands.

```
Started by user salvad basha shaik
Running as SYSTEM
Building on the built-in node in workspace /var/lib/jenkins/workspace/package_kubernetes_v2
The recommended git tool is: NONE
No credentials specified
> git rev-parse --resolve-git-dir /var/lib/jenkins/workspace/package_kubernetes_v2.git # timeout=10
Fetching changes from the remote Git repository
> git config remote.origin.url https://github.com/salvathshaik/final-devops-project.git # timeout=10
Fetching upstream changes from https://github.com/salvathshaik/final-devops-project.git
> git -version # timeout=10
> git -version # 'git' version 2.17.1'
> git fetch --tags --progress .. https://github.com/salvathshaik/final-devops-project.git +refs/heads/*:refs/remotes/origin/* # timeout=10
> git rev-parse refs/remotes/origin/main^{commit} # timeout=10
Checking out Revision 2b1ae87abc06f6576dc75df28620816ecf9ded (refs/remotes/origin/main)
> git config core.sparsecheckout # timeout=10
> git checkout -f 2b1ae87abc06f6576dc75df28620816ecf9ded # timeout=10
Commit message: "added hosts file in version2"
> git rev-list --no-walk 2b1ae87abc06f6576dc75df28620816ecf9ded # timeout=10
[package_kubernetes_v2] $ /bin/sh -xe /tmp/jenkins11287869471643616226.sh
+ sudo docker login -u salvathshaik -p Allah786000
WARNING! Using --password via the CLI is insecure. Use --password-stdin.
WARNING! Your password will be stored unencrypted in /var/lib/jenkins/.docker/config.json.
```

Docker login and docker push has been started

Deploying to kubernetes has been started

```
package_kubernetes_v2 #7 Console [Jenkins] - Mozilla Firefox
package_kubernetes_v2 x package_kubernetes_v2 x localhost:8080/ABCtech x localhost:32768/ABCtech x task 4 version2 - salvat... x +
localhost:8080/job/package_kubernetes_v2/7/console
Dashboard > package_kubernetes_v2 > #7

        },
        "failed": false
    }
}

TASK [Create a Deployment by reading the definition from a local file] *****
changed: [localhost]

TASK [debug] *****
ok: [localhost] => {
    "out": {
        "changed": true,
        "cmd": [
            "kubectl",
            "-c=kubeconfig=/etc/kubernetes/admin.conf",
            "apply",
            "-f",
            "/home/edureka/Desktop/deployment.yml"
        ],
        "delta": "0:00:00.46711",
        "end": "2022-11-18 12:33:18.888889",
        "failed": false,
        "rc": 0,
        "start": "2022-11-18 12:33:18.421278",
        "stderr": "",
        "stderr_lines": [],
        "stdout": "deployment.apps/abctechnologies-dep unchanged\nservice/abc-tech-service unchanged",
        "stdout_lines": [
            "deployment.apps/abctechnologies-dep unchanged",
            "service/abc-tech-service unchanged"
        ]
    }
}
```

Pacakge_kubernetes_v2 job got success message

The screenshot shows a Linux desktop environment with a terminal window and a browser window.

The terminal window displays Jenkins logs for a job named "package_kubernetes_v2". The logs show a successful deployment command:

```
..-kubeconfig=/etc/kubernetes/admin.conf",
"apply",
"-f",
"/home/edureka/Desktop/deployment.yaml"
],
"delta": "0:00:00.467611",
"end": "2022-11-18 12:33:18.888889",
"failed": false,
"rc": 0,
"start": "2022-11-18 12:33:18.421278",
"stderr": "",
"stderr_lines": [],
"stdout": "deployment.apps/abctechnologies-dep unchanged\nservice/abc-tech-service unchanged",
"stdout_lines": [
"deployment.apps/abctechnologies-dep unchanged",
"service/abc-tech-service unchanged"
]
}
```

PLAY RECAP *****
localhost : ok=7 changed=1 unreachable=0 failed=0 skipped=0 rescued=0 ignored=0

Finished: SUCCESS

The browser window shows the Jenkins job console for "package_kubernetes_v2" with the status "SUCCESS".

Docker image validation in edureka local system

The screenshot shows a Linux desktop environment with a terminal window.

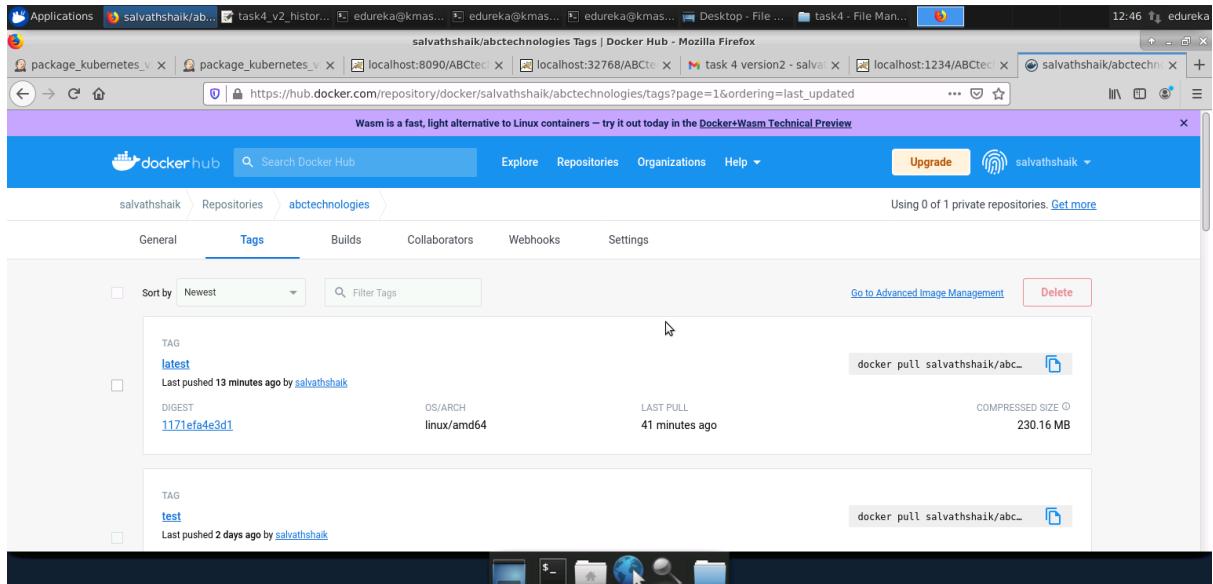
The terminal window displays the output of a Docker image validation command:

```
edureka@kmaster:~$ #validating the docker image and docker containers running or not
edureka@kmaster:~$ docker images
```

REPOSITORY	TAG	IMAGE ID	CREATED	SIZE
salvathsraik/abctechnologies	latest	f489fff6d652	41 hours ago	558MB
salvathsraik/abctechnologies	<none>	261179221bf8	41 hours ago	558MB
salvathsraik/abctechnologies	<none>	0c9322c16df2	42 hours ago	558MB
salvathsraik/abctechnologies	<none>	bcc1f1beaf8f	42 hours ago	558MB
salvathsraik/abctechnologies	<none>	e0726d61ee92	10 days ago	558MB
<none>	<none>	a33a265f1ae1	10 days ago	518MB
rancher/mirrored-flannelcni-flannel	v0.20.1	d66192101c64	2 weeks ago	59.4MB
ubuntu	latest	a8780b506fa4	2 weeks ago	77.8MB
rancher/mirrored-flannelcni-flannel-cni-plugin	v1.1.0	fcecffc7ad4a	6 months ago	8.09MB
k8s.gcr.io/kube-proxy	v1.18.20	27ff8bb5d1985	17 months ago	117MB
k8s.gcr.io/kube-apiserver	v1.18.20	7d8d2960de69	17 months ago	173MB
k8s.gcr.io/kube-controller-manager	v1.18.20	e7c545a60706	17 months ago	162MB
k8s.gcr.io/kube-scheduler	v1.18.20	a05ala79adaa	17 months ago	96.1MB
k8s.gcr.io/pause	3.2	80d28bedfe5d	2 years ago	683kB
k8s.gcr.io/coredns	1.6.7	67da37a9a360	2 years ago	43.8MB
k8s.gcr.io/etcd	3.4.3-0	303ce5db0e90	3 years ago	288MB

```
edureka@kmaster:~$
```

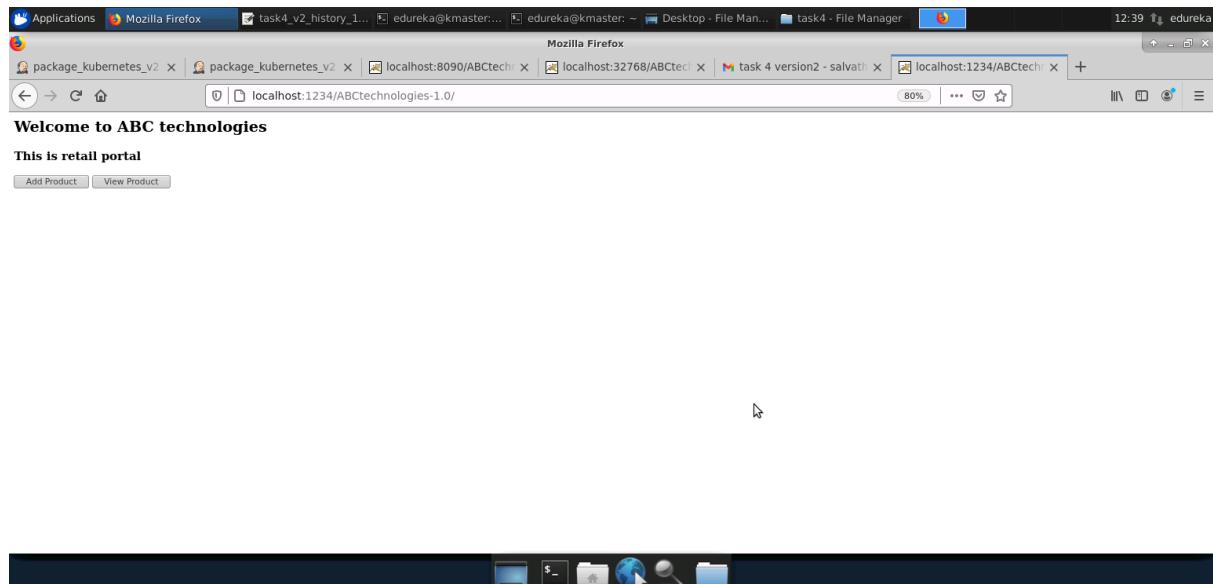
Docker hub push validation



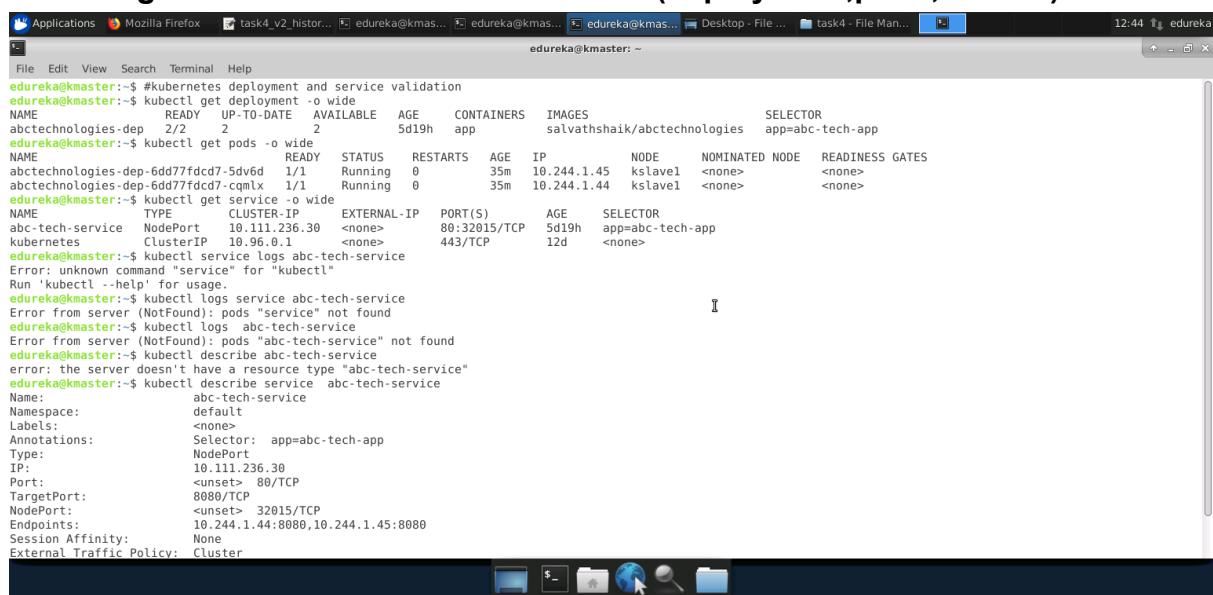
Docker container validation

File	Edit	View	Search	Terminal	Help		edureka@kmaster: ~
k8s.gcr.io/etcd						3.4.3-0	303ce5db0e90 3 years ago 288MB
edureka@kmaster:~\$ docker ps							
CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES	
d61d10190bd4	salvathkar/abctechnologies	"bin/sh -c '/usr/local/bin/etcd --conf /etc...'	32 minutes ago	Up 32 minutes	0.0.0.0:1234->8080/tcp	abc-application	
af9020956161	67da37a9a360	"coredns -conf /etc..."	54 minutes ago	Up 54 minutes		k8s_coredns_coredns-66bf46f46	
7fb-q6q8x_kube-system	1f0b63d3-068d-4075-a7e-9590b085ef2a	"pause"	54 minutes ago	Up 54 minutes		k8s_POD_coredns-66bf46f7f8-	
4f73be5b41	k8s.gcr.io/pause:3.2	"pause"	54 minutes ago	Up 54 minutes		k8s_coredns_coredns-66bf46f46	
bg6q8x_kube-system	1f0b63d3-068d-4075-a7e-9590b085ef2a	"coredns -conf /etc..."	55 minutes ago	Up 54 minutes		k8s_POD_coredns-66bf46f7f8-	
7fbcd9b6c53b	67da37a9a360	"coredns -conf /etc..."	55 minutes ago	Up 54 minutes		k8s_coredns_coredns-66bf46f46	
7f8-4sxtk_kube-system	3858cc7-7461-495-84db-37619107233f	"pause"	55 minutes ago	Up 55 minutes		k8s_POD_coredns-66bf46f7f8-	
4sxtk_kube-system	3858cc7-7461-495-84db-37619107233f	"opt/bin/flanneld ..."	55 minutes ago	Up 55 minutes		k8s_kube-flannel_kube-flann	
a9f4a1412ee3	d66192101c64	"el-ds-xr4m_kube-flannel_4fd68a30-73f7-436b-85a5-291ac8ad63bb	23	Up 55 minutes			
c9e32e5ebd13	27ff88d51985	"usr/local/bin/kube-e..."	55 minutes ago	Up 55 minutes		k8s_kube-proxy_kube-proxy-r	
bfw8_kube-system	bb7d5397-506d-4916-bfb7-7344a8292b06	"pause"	55 minutes ago	Up 55 minutes		k8s_POD_kube-flannel-ds-xr4	
af85fc46e1a1	k8s.gcr.io/pause:3.2	"pause"	55 minutes ago	Up 55 minutes			
mr_kube-flannel_4fd68a30-73f7-436b-85a5-291ac8ad63bb	22	"etcd --advertise-cl..."	55 minutes ago	Up 55 minutes			
5e4e2a272f36	k8s.gcr.io/pause:3.2	"pause"	55 minutes ago	Up 55 minutes			
be-system_bbd75397-5d66-4916-bfb7-7344a8292b06	21	"kube-controller-man..."	56 minutes ago	Up 56 minutes			
12b2caef9f052	e7c545a60706	"kube-controller-manager-kmaster_kube-system_2cf16e7db66397e307ead96f5c8a447	21	Up 56 minutes			
kube-controller-manager-kmaster_kube-system_2cf16e7db66397e307ead96f5c8a447	21	"kube-apiserver --ad..."	56 minutes ago	Up 56 minutes			
40139597877	7d8d2960e69	"server-kmaster_kube-system_6b062f177490920443c97cd3a090e66b	22	Up 56 minutes			
7bd83e7a4e1	303ce5db0e90	"etcd --advertise-cl..."	56 minutes ago	Up 56 minutes			
system_a4f1cedd69431ed65bf4d3-49f6111c5	21	"kube-scheduler --au..."	56 minutes ago	Up 56 minutes			
a759ef9d377f	a65a1a7940d	"eduler-kmaster_kube-system_e541886a4cda0424b9879a78869adc51	22	Up 56 minutes			
91f87eb3e9bf	k8s.gcr.io/pause:3.2	"pause"	56 minutes ago	Up 56 minutes			
ter_kube-system_6b062f177490920443c97cd3a090e66b	22	"kube-apiserver --ad..."	56 minutes ago	Up 56 minutes			
69f1ae86f05	k8s.gcr.io/pause:3.2	"pause"	56 minutes ago	Up 56 minutes			
system_a4f1cedd69431ed65bf4d3-49f6111c5	21	"etcd --advertise-cl..."	56 minutes ago	Up 56 minutes			
60a339209b49	k8s.gcr.io/pause:3.2	"pause"	56 minutes ago	Up 56 minutes			
ter_kube-system_e541886a4cda0424b9879a78869adc51	22	"kube-scheduler --au..."	56 minutes ago	Up 56 minutes			

Docker container validation in browser



Validating kubernetes validation resources(deployment,pods,service)



Validating the kubernetes container with service port

Welcome to ABC technologies

This is retail portal

Add Product View Product

Creating the pipeline(CI_CD_PIPELINE_KUBERNETES_V2)

Dashboard > salwad basha shaik > My Views

↑ People

👤 Status

💻 Builds

⚙️ Configure

🔗 My Views

🔑 Credentials

View name

CI_CD_PIPELINE_KUBERNETES_V2

Type

Build Pipeline View

Shows the jobs in a build pipeline view. The complete pipeline of jobs that a version propagates through are shown as a row in the view.

Include a global view

Shows the content of a global view.

List View

Shows items in a simple list format. You can choose which jobs are to be displayed in which view.

My View

This view automatically displays all the jobs that the current user has an access to.

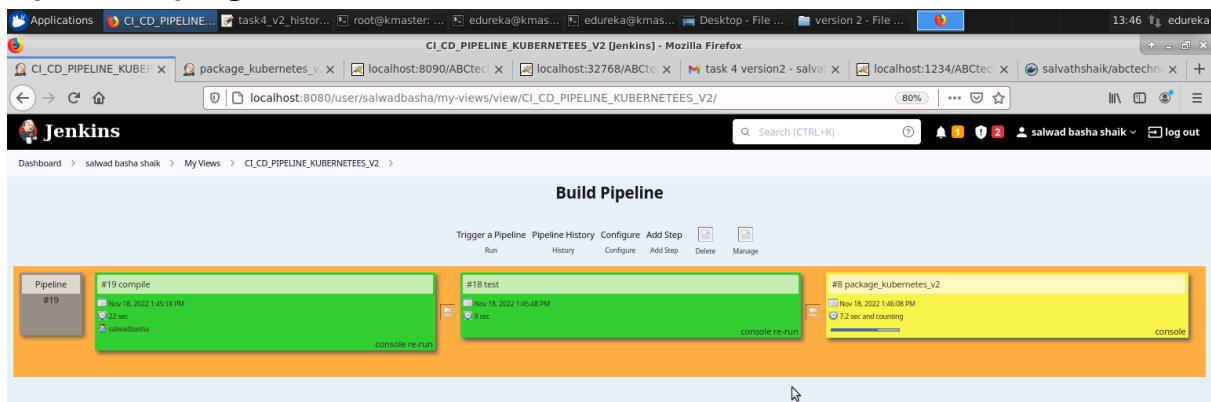
Modifying the 'package_kubernetes_v2' job to watch the test job and trigger when it completes.

The screenshot shows the Jenkins configuration page for the 'package_kubernetes_v2' job. The left sidebar has 'Build Triggers' selected. Under 'Build Triggers', the 'Build after other projects are built' checkbox is checked. In the 'Projects to watch' field, 'test' is listed. Below this, several other trigger options are available but not selected: 'Trigger only if build is stable', 'Trigger even if the build is unstable', 'Trigger even if the build fails', 'Always trigger, even if the build is aborted', 'Build periodically', 'GitHub hook trigger for GITScm polling', and 'Poll SCM'. At the bottom are 'Save' and 'Apply' buttons.

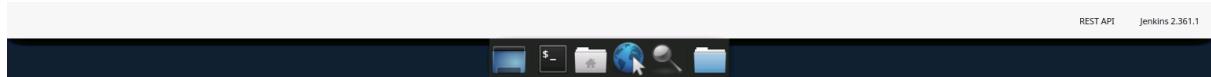
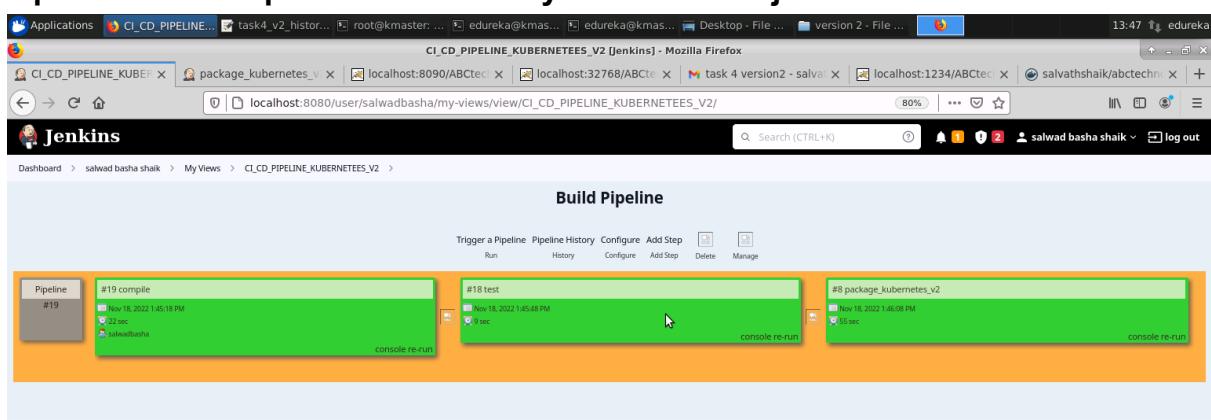
Pipeline started and then compile job has triggered

The screenshot shows the Jenkins pipeline view for 'CI_CD_PIPELINE_V2'. A pipeline item labeled '#19' is shown, with a status message: 'Nov 18, 2022 1:45:18 PM 13 sec and counting salwadasha'. Below this is a 'console' link. To the right, a 'test' job is shown with two builds: '#19' and '#20'. Further to the right, a 'package_kubernetes_v2' job is shown with two builds: '#19' and '#20'. At the top, there are tabs for 'Trigger a Pipeline', 'Pipeline History', 'Configure', 'Add Step', 'Run', 'History', 'Configure', 'Add Step', 'Delete', and 'Manage'. At the bottom, there are links for 'REST API' and 'Jenkins 2.361.1'.

Pipeline in-progress



Pipeline has completed successfully with all three jobs



Validating package_kubernetes_v2 job status stating who trigger(started by upstream project 'test')

The screenshot shows the Jenkins interface for a build named 'package_kubernetes_v2' (Build #8). The build was triggered by an upstream project named 'test'. The status is green, indicating success. The build was started 1 min 22 sec ago and took 55 sec on built-in. The build log indicates no changes were made.

Build #8 (Nov 18, 2022, 1:46:08 PM)

Started upstream project [test](#) build number 18 originally caused by:

- Started by upstream project [compile](#) build number 19 originally caused by:
 - Started by user [salwad basha shaik](#)

Revision: 2b1aeb7aabca6f6576d75df28620816eect9ded
Repository: <https://github.com/salvathshaik/final-dev-ops-project.git>
refs/remotes/origin/main

REST API Jenkins 2.361.1

Package_kubernetes_v2 job success log

The screenshot shows the Jenkins console output for the 'package_kubernetes_v2' job. The log output shows the deployment of a Kubernetes deployment named 'abc-technology-dep' with a service named 'abc-tech-service'. The deployment was successful, indicated by the status 'ok=7' in the play recap.

```
--> kubeconfig=/etc/kubernetes/admin.conf
"apply",
"-f",
"/home/edureka/Desktop/deployment.yaml"
],
"delta": "0:00:04.890799",
"end": "2022-11-18 13:47:03.822945",
"failed": false,
"rc": 0,
"start": "2022-11-18 13:46:58.932146",
"stderr": "",
"stderr_lines": [],
"stdout": "deployment.apps/abctechnologies-dep unchanged\nservice/abc-tech-service unchanged",
"stdout_lines": [
"deployment.apps/abctechnologies-dep unchanged",
"service/abc-tech-service unchanged"
]
}
}

PLAY RECAP ****
localhost          : ok=7   changed=2   unreachable=0   failed=0    skipped=0   rescued=0   ignored=0

Finished: SUCCESS
```

REST API Jenkins 2.361.1

Docker resources validation:

```

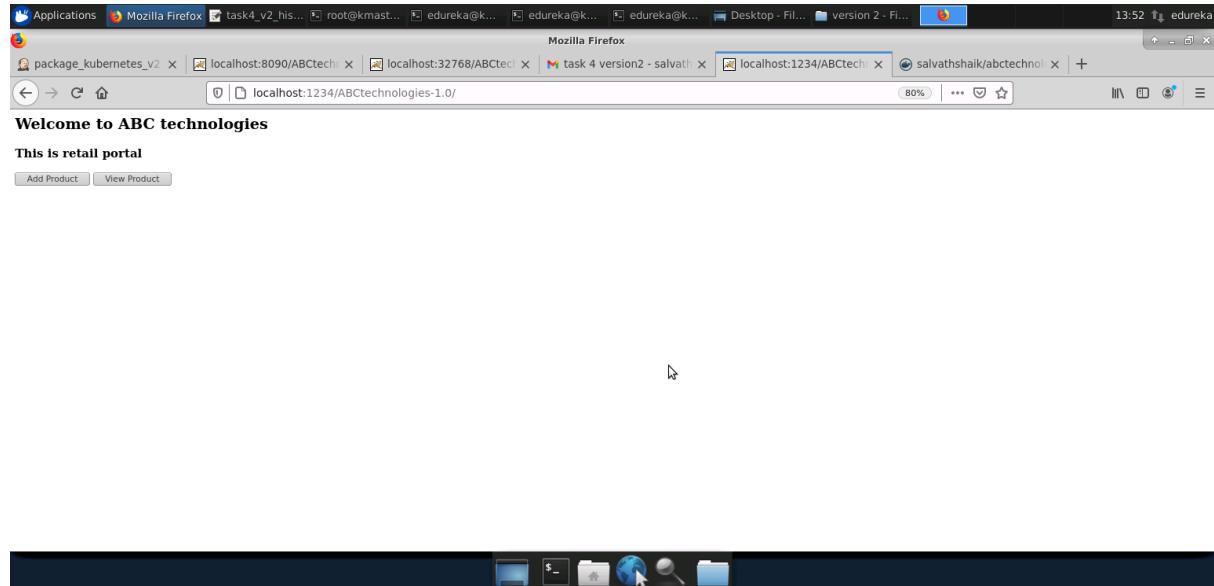
edureka@kmaster:~$ docker images
REPOSITORY          TAG      IMAGE ID      CREATED       SIZE
salvathshaik/abctechnologies    latest   a682f79b577  17 minutes ago  558MB
salvathshaik/abctechnologies    <none>   7d5e72a8f7a7  27 minutes ago  558MB
salvathshaik/abctechnologies    <none>   4a2d02a0f58c  28 minutes ago  558MB
salvathshaik/abctechnologies    <none>   3d345b41ee1a  39 minutes ago  558MB
salvathshaik/abctechnologies    <none>   c28e226b1842  48 minutes ago  558MB
salvathshaik/abctechnologies    <none>   f489fff6fd652 42 hours ago   558MB
salvathshaik/abctechnologies    <none>   261179221bf8  42 hours ago   558MB
salvathshaik/abctechnologies    <none>   0c9322c16df2  43 hours ago   558MB
salvathshaik/abctechnologies    <none>   bcc1f1beaf8f  43 hours ago   558MB
salvathshaik/abctechnologies    <none>   e0726d61ee92  10 days ago   558MB
<none>                  <none>   a33a265f1ae1  10 days ago   518MB
rancher/mirrored-flannelcni-flannel  v0.29.1  d66192101c64  2 weeks ago   58.4MB
ubuntu                  latest   a8780b5067a4  2 weeks ago   77.8MB
rancher/mirrored-flannelcni-flannel-cni-plugin v1.1.0   fcecffc7ad4a  6 months ago   8.09MB
k8s.gcr.io/kube-proxy          v1.18.20  27ffbb0d1985  17 months ago  117MB
k8s.gcr.io/kube-apiserver     v1.18.20  7d8d2960de69  17 months ago  173MB
k8s.gcr.io/kube-scheduler      v1.18.20  a05ala79adaa  17 months ago  96.1MB
k8s.gcr.io/kube-controller-manager v1.18.20  e7c545a60706  17 months ago  162MB
k8s.gcr.io/pause              3.2      88d28bedfe5d  2 years ago   683kB
k8s.gcr.io/coredns            1.6.7    67da37a9a360  2 years ago   43.8MB
k8s.gcr.io/etcd               3.4.3-0   383ce5db0e99  3 years ago   288MB
edureka@kmaster:~$ docker ps
CONTAINER ID        IMAGE               COMMAND             CREATED            STATUS              PORTS               NAMES
f53393727808        salvathshaik/abctechnologies   "/bin/sh -c '/usr/lo..."   3 minutes ago     Up 3 minutes        0.0.0.0:1234->8080/tcp   abc-application
0929f522861e        salvathshaik/abctechnologies   "/bin/sh -c '/usr/lo..."   17 minutes ago    Up 17 minutes       0.0.0.0:32772->8080/tcp  suspicious_chaplygin
2faaab8f0d0a        7d5e72a8f7a7           "/bin/sh -c '/usr/lo..."   27 minutes ago    Up 27 minutes       0.0.0.0:32771->8080/tcp  zealous_dhawan
7ef5a7145568        4a2d02a0f58c           "/bin/sh -c '/usr/lo..."   28 minutes ago    Up 28 minutes       0.0.0.0:32770->8080/tcp  flamboyant_dijkstra
2dc86110048c        3d345b41ee1a           "/bin/sh -c '/usr/lo..."   38 minutes ago    Up 38 minutes       0.0.0.0:32769->8080/tcp  ecstatic_hamilton
ae4317a5e0dc        c28e226b1842           "/bin/sh -c '/usr/lo..."   48 minutes ago    Up 48 minutes       0.0.0.0:32768->8080/tcp  busy_wu
af9820956161        67da37a9a360           "/coredns -conf /etc..."  2 hours ago       Up 2 hours          0.0.0.0:53->53/tcp       k8s_coredns_coredns-66bff467f8
67f8-bqg8x_kube-system_1fb6b3d3-068d-4075-a47e-9590b085efaf2_21  7d7a37a9a360           "/pause"           2 hours ago       Up 2 hours          0.0.0.0:4467f8->4467f8
47fe3be5bb41        k8s.gcr.io/pause:3.2        "/pause"           2 hours ago       Up 2 hours          0.0.0.0:4467f8->4467f8

```

Docker hub validation

The screenshot shows a Docker Hub repository page for the user 'salvathshaik'. The repository name is 'abctechnologies'. The 'Tags' tab is selected, displaying two tags: 'latest' and 'test'. The 'latest' tag was last pushed 5 minutes ago by 'salvathshaik'. The 'test' tag was last pushed 2 days ago by 'salvathshaik'. Both tags have a digest of '2ce634e8bff4', OS/ARCH of 'linux/amd64', and a compressed size of 230.16 MB. There are download and delete buttons for each tag.

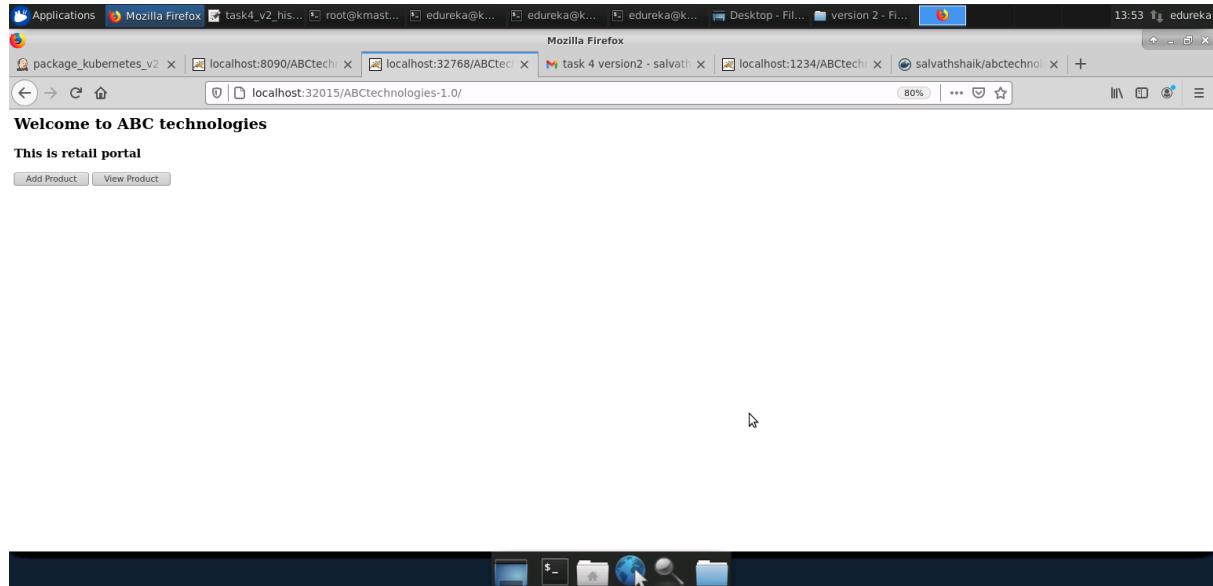
Docker container port validation in browser



Kubernetes resources validation

```
File Edit View Search Terminal Help
edureka@kmaster: ~
rbfw8 kube-system_bbd75397-5d66-4916-bfbd-7344a8292b06_21
af5fc46e1e k8s.gcr.io/pause:3.2 "/pause"
4m7 kube-flannel_4fd68a30-73f7-436b-85a5-291aca8d63bb_22
3e4e32a72f36 k8s.gcr.io/pause:3.2 "/pause"
ube-system bbd75397-5d66-4916-bfbd-7344a8292b06_21
12bc2ae9f052 e7c545a60706 "kube-controller-man..." 2 hours ago Up 2 hours
r kube-controller-manager-kmaster kube-system_2cf1e67db6397e307ead965fc8a447_21
401395a978f7 78d82960de69 "kube-apiserver --ad..." 2 hours ago Up 2 hours
iserver-kmaster kube-system_6b062f177409020443c97cd3a009ee6b_22
fdbb3e7ade1 303ce5d0e90 "etcd --advertise-cl..." 2 hours ago Up 2 hours
-system a4e1cedd69431ed65bf4d3c49f611c5_21
a759ef9d37ff a05a1a79ada "kube-scheduler --au..." 2 hours ago Up 2 hours
heder-kmaster kube-system_e541886a4cd0424b9879a78869adc51_22
91f87eb3e9bf k8s.gcr.io/pause:3.2 "/pause"
ster_kube-system_6b62f177409020443c97cd3a009ee6b_22
69f1ae86f0f5 k8s.gcr.io/pause:3.2 "/pause"
system_a4e1cedd69431ed65bf4d3c49f611c5_21
60a339209b4e k8s.gcr.io/pause:3.2 "/pause"
ster_kube-system_e541886a4cd0424b9879a78869adc51_22
70dcf6f9c2f2 k8s.gcr.io/pause:3.2 "/pause"
nager-kmaster kube-system_2cf1e67db6397e307ead965fc8a447_21
edureka@kmaster: $ kubectl get deployment -o wide
NAME READY UP-TO-DATE AVAILABLE AGE CONTAINERS IMAGES SELECTOR
abctechnologies-dep 2/2 2 2 5d20h app salvathshaik/abctechnologies app=abc-tech-app
edureka@kmaster: $ kubectl get pods -o wide
NAME READY STATUS RESTARTS AGE IP NODE NOMINATED NODE READINESS GATES
abctechnologies-dep-6dd77fdc7-5dv6d 1/1 Running 0 187m 10.244.1.45 kslavel1 <none> <none>
abctechnologies-dep-6dd77fdc7-cqnlx 1/1 Running 0 187m 10.244.1.44 kslavel1 <none> <none>
edureka@kmaster: $ kubectl get service -o wide
NAME TYPE CLUSTER-IP EXTERNAL-IP PORT(S) AGE SELECTOR
abc-tech-service NodePort 10.111.236.30 <none> 80:32915/TCP 5d20h app=abc-tech-app
kubernetes ClusterIP 10.96.0.1 <none> 443/TCP 12d <none>
edureka@kmaster:~$
```

Kubernetes service port validation in browser



References i have followed:

https://docs.ansible.com/ansible/latest/collections/community/docker/docker_image_module.html
https://docs.ansible.com/ansible/latest/collections/community/docker/docker_container_module.html
<https://github.com/ansible-collections/community.docker>
<https://pypi.org/project/docker/>
<https://pypi.org/project/docker-py/>
<https://github.com/ansible-collections/community.docker>
<https://docker-py.readthedocs.io/en/latest/>
<https://www.baeldung.com/ops/docker-removing-images#:~:text=Forcefully%20Remove%20Containers%20and%20Images&text=The%20%2Df%20flag%20is%20used,the%20running%20Docker%20containers%20forcefully.&text=The%20docker%20images%20%2Dqa%20will,forcefully%20remove%20the%20Docker%20image>
<https://stackoverflow.com/questions/22769568/system-specific-variables-in-ansible>
<https://stackoverflow.com/questions/60834692/how-to-completely-remove-ansible-2-8-3-on-ubuntu-18-04>
--
https://docs.ansible.com/ansible/2.6/modules/k8s_module.html#examples
https://docs.ansible.com/ansible/latest/user_guide/guide_rolling_upgrade.html
<https://adamtheautomator.com/ansible-kubernetes/>
<https://adamtheautomator.com/ansible-template/>
`ansible-playbook sample-playbook.yml -e 'ansible_python_interpreter=/usr/bin/python3'`
https://docs.ansible.com/ansible/latest/reference_appendices/python_3_support.html
https://docs.ansible.com/ansible/2.9/modules/k8s_service_module.html

<https://github.com/DeekshithSN/sample-web-application> -
<https://www.youtube.com/watch?v=NSk0NHkTjDs>
<https://www.baeldung.com/ops/jenkins-parameterized-builds>
<https://stackoverflow.com/questions/56003777/how-to-pass-environment-variable-in-kubectl-deployment>
<https://stackoverflow.com/questions/62108860/kubectl-no-matches-for-kind-service-in-version-apps-v1>
<https://stackoverflow.com/questions/59579482/kubectl-apply-error-from-server-forbidden-authentication-required-jenkins>
<https://stackoverflow.com/questions/51489359/docker-using-password-via-the-cli-is-insecure-use-password-stdin>
<https://kubernetes.io/docs/reference/kubectl/cheatsheet/>

Task 5: Using Prometheus, monitor the resources like CPU utilization: Total Usage, Usage per core, usage breakdown, memory, and network on the instance by providing the endpoints on the local host. Install the node exporter and add the URL to the target in Prometheus. Using this data, log in to Grafana and create a dashboard to show the metrics.

Approach i have followed:

First of all we need to have prometheus,Grafana but edureka lab already installed these tools. So I had installed the node-exporter in master node to monitor the metrics of a node and added the target url in the prometheus.yml file so that prometheus will start to monitor.

And I have monitored and captured the CPU,memory and network of the target node. And I have created a dashboard by selecting the prometheus app from Grafana and then created a Panel for visualizing the metrics in Grafana.

After installing node_exporter I have added the below job in the prometheus.yml file to get the metrics from the target.

```
-----
- job_name: 'node_exporter'
  scrape_interval: 5s
  static_configs:
    - targets: ['localhost:9100']
-----
/opt/prometheus-2.27/prometheus.yml
# my global config
global:
  scrape_interval: 15s # Set the scrape interval to every 15 seconds. Default is
every 1 minute.
  evaluation_interval: 15s # Evaluate rules every 15 seconds. The default is every 1
minute.
  # scrape_timeout is set to the global default (10s).

# Alertmanager configuration
alerting:
  alertmanagers:
    - static_configs:
      - targets:
        # - alertmanager:9093

# Load rules once and periodically evaluate them according to the global
'evaluation_interval'.
rule_files:
```

```

# - "first_rules.yml"
# - "second_rules.yml"

# A scrape configuration containing exactly one endpoint to scrape:
# Here it's Prometheus itself.
scrape_configs:
  # The job name is added as a label `job=<job_name>` to any timeseries scraped
  # from this config.
  - job_name: 'prometheus'

    # metrics_path defaults to '/metrics'
    # scheme defaults to 'http'.

    static_configs:
      - targets: ['localhost:9090']

      - job_name: 'node_exporter'
        scrape_interval: 5s
        static_configs:
          - targets: ['localhost:9100']

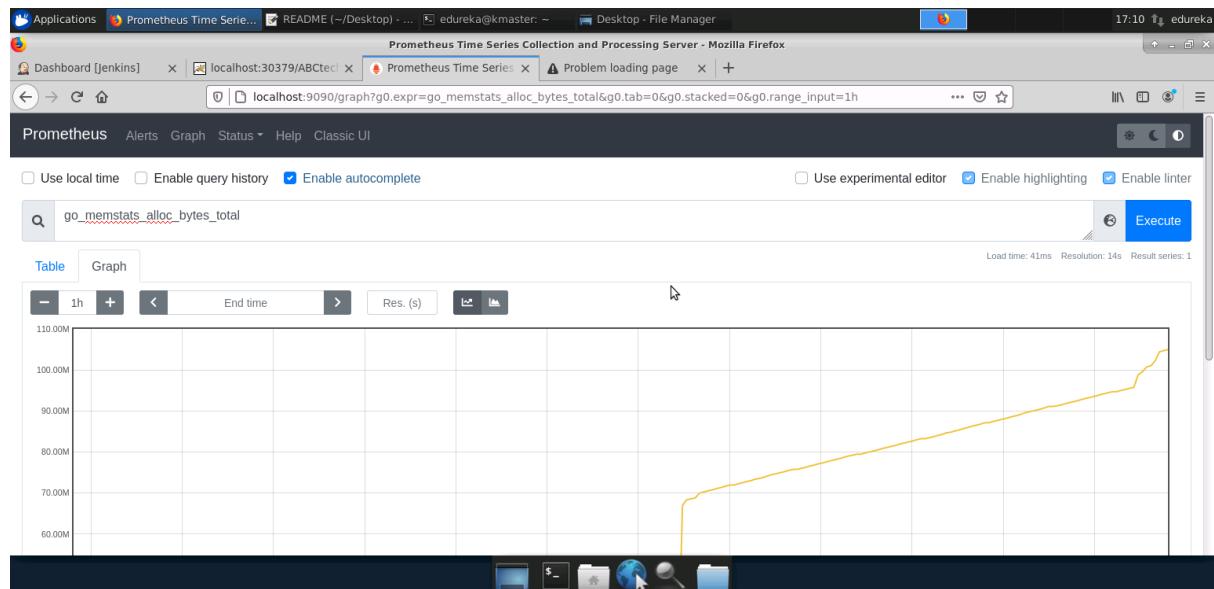
```

Please find the screenshots:

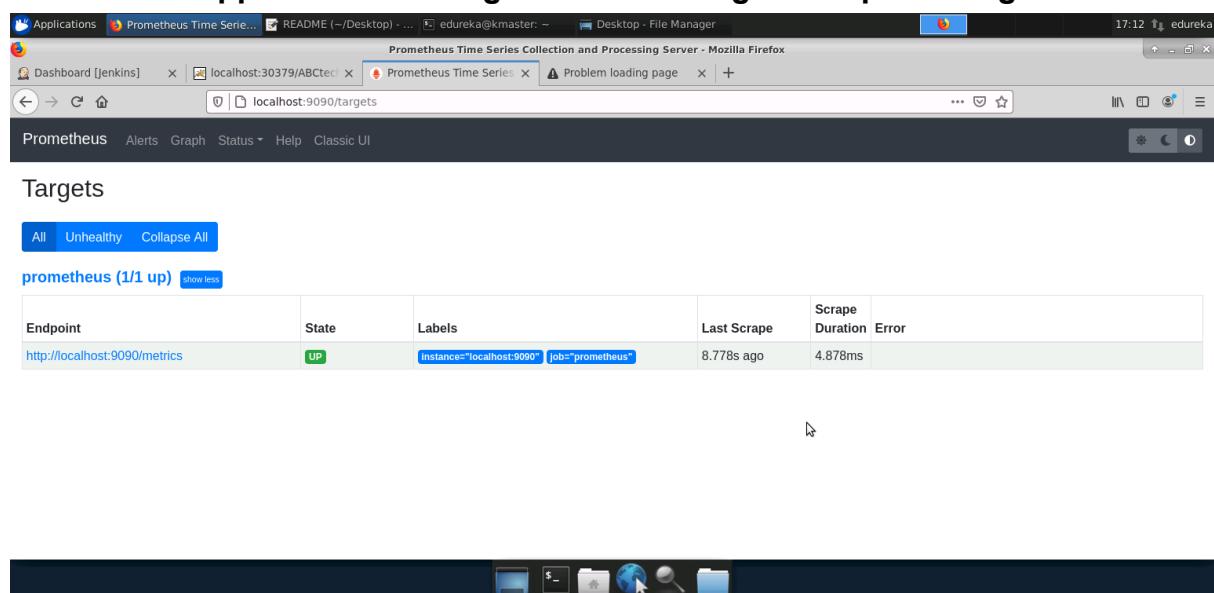
Prometheus self monitoring

The screenshot shows the Prometheus UI interface. At the top, there are several tabs: Applications, Prometheus Time Serie..., README (~/Desktop) ..., edureka@kmaster: ~, Desktop - File Manager, and a Firefox window titled 'Prometheus Time Series Collection and Processing Server - Mozilla Firefox'. The main content area displays a query result for the metric 'go_memstats_alloc_bytes_total'. The query is: `go_memstats_alloc_bytes_total{instance="localhost:9090", job="prometheus"}`. The result shows a single data point: 192514240. Below the result, there are buttons for 'Add Panel' and 'Remove Panel'. At the bottom of the screen, there is a dark dock bar with various icons.

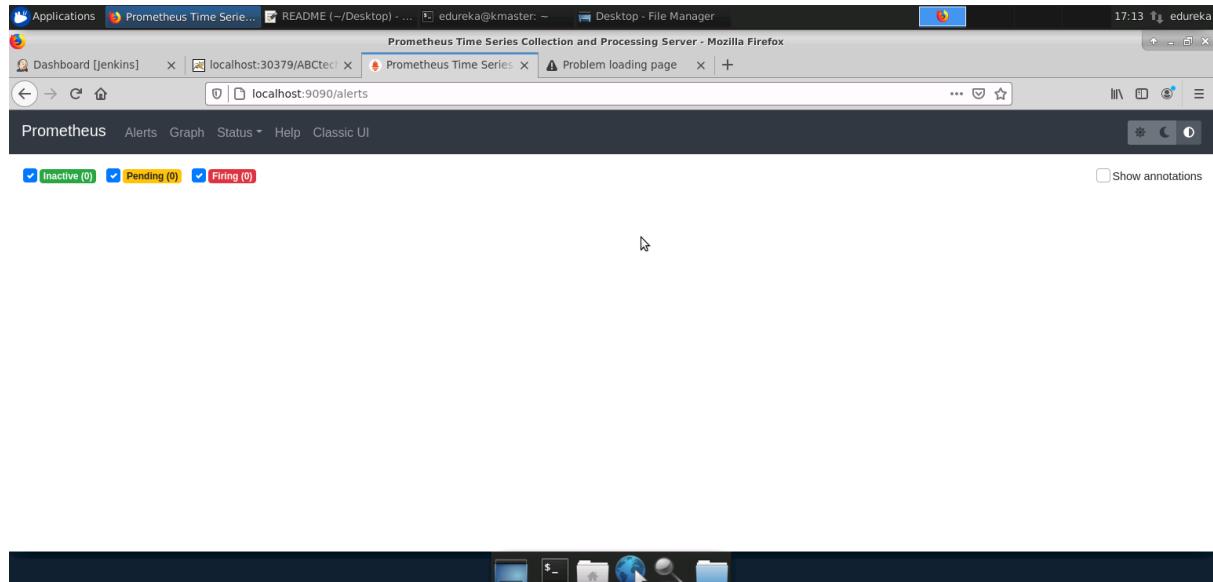
Graph visualization for the metrics in prometheus



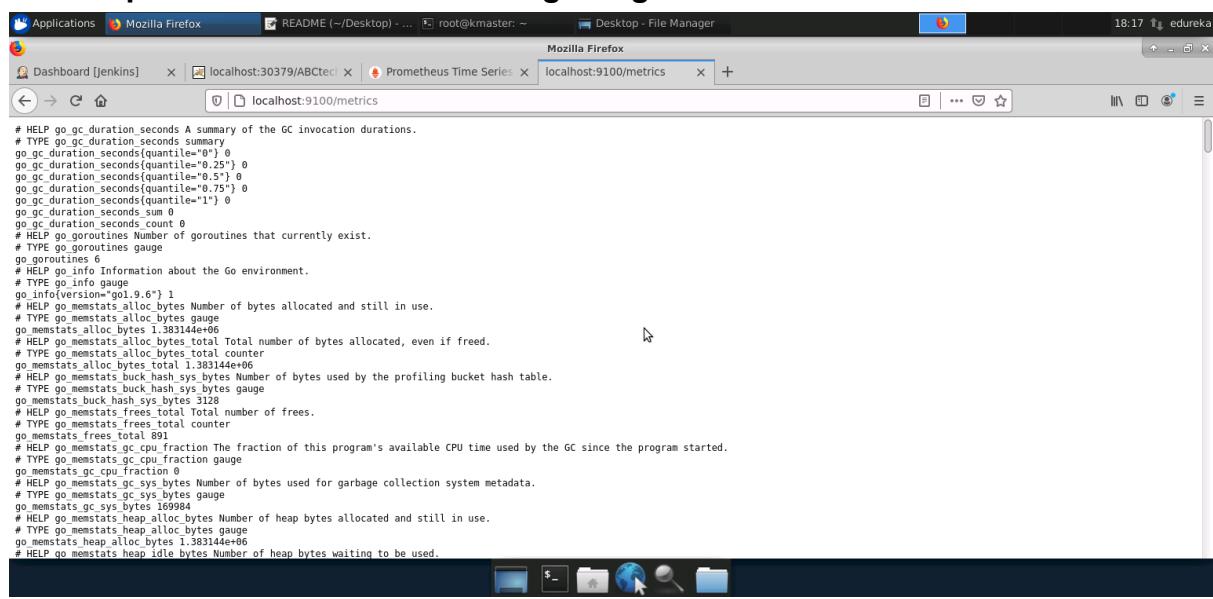
Prometheus application self target before adding our required target



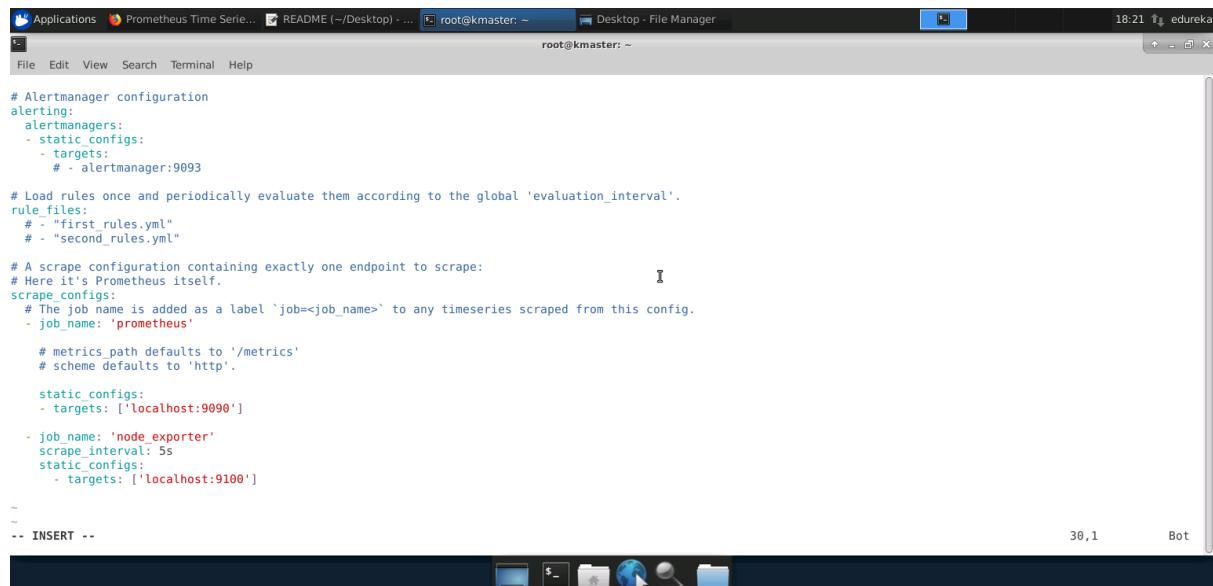
Alerts tab: If we configured alertmanager it will list the alerts here



node exporter data validation through target



added_target in prometheus file



```
# Alertmanager configuration
alerting:
  alertmanagers:
    - static_configs:
      - targets:
        # - alertmanager:9093

# Load rules once and periodically evaluate them according to the global 'evaluation_interval'.
rule_files:
  # - "first_rules.yml"
  # - "second_rules.yml"

# A scrape configuration containing exactly one endpoint to scrape:
# Here it's Prometheus itself.
scrape_configs:
  # The job name is added as a label `job=<job_name>` to any timeseries scraped from this config.
  - job_name: 'prometheus'

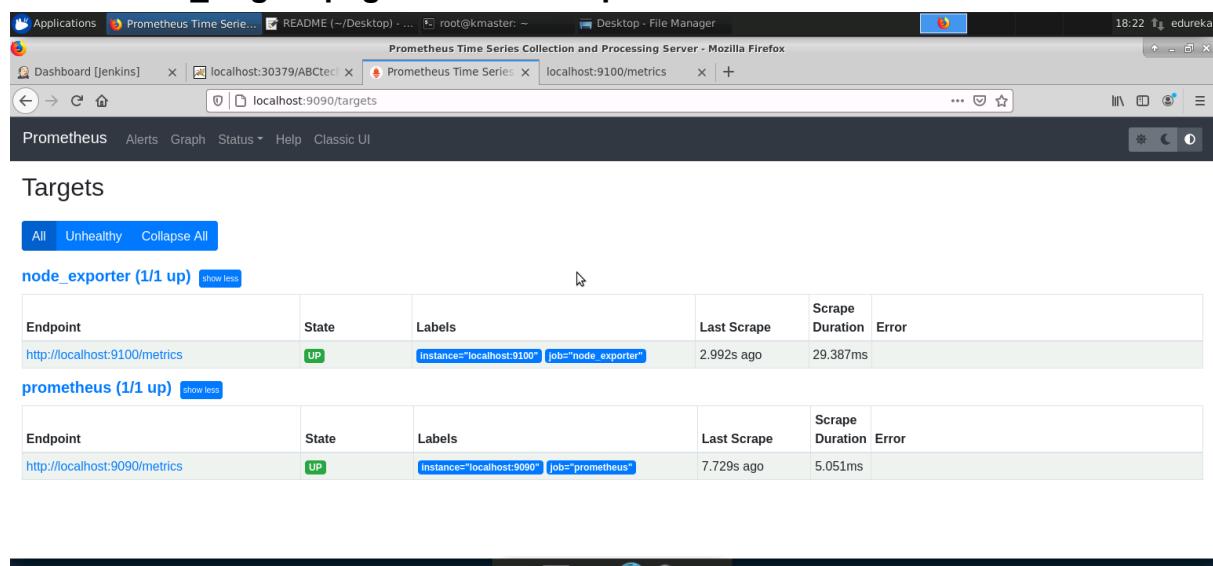
    # metrics_path defaults to '/metrics'
    # scheme defaults to 'http'.

    static_configs:
      - targets: ['localhost:9090']

  - job_name: 'node_exporter'
    scrape_interval: 5s
    static_configs:
      - targets: ['localhost:9100']

-- INSERT --
```

Prometheus_targets page with node exporter of a node



The screenshot shows a Mozilla Firefox browser window displaying the Prometheus Targets page. The URL in the address bar is `localhost:9090/targets`. The page header includes the Prometheus logo and navigation links for Dashboard, Jenkins, Prometheus Time Series Collection and Processing Server, localhost:30379/ABCtel, localhost:9100/metrics, and localhost:9090/targets.

The main content area displays two tables of targets:

Endpoint	State	Labels	Last Scrape	Scrape Duration	Error
http://localhost:9100/metrics	UP	instance="localhost:9100" job="node_exporter"	2.992s ago	29.387ms	

Endpoint	State	Labels	Last Scrape	Scrape Duration	Error
http://localhost:9090/metrics	UP	instance="localhost:9090" job="prometheus"	7.729s ago	5.051ms	

cpu usage breakdown for node

The screenshot shows the Prometheus UI with the query `node_cpu_seconds_total`. The results are displayed in a table:

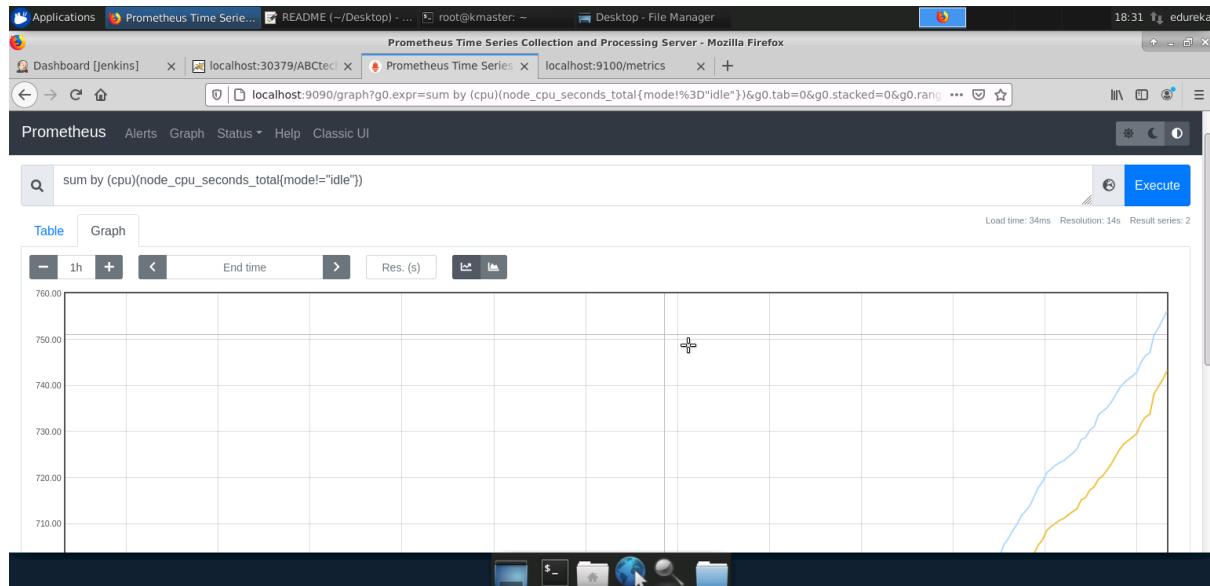
Series	Value
node_cpu_seconds_total(cpu="0", instance="localhost:9100", job="node_exporter", mode="idle")	6736.82
node_cpu_seconds_total(cpu="0", instance="localhost:9100", job="node_exporter", mode="iowait")	95.05
node_cpu_seconds_total(cpu="0", instance="localhost:9100", job="node_exporter", mode="io")	0
node_cpu_seconds_total(cpu="0", instance="localhost:9100", job="node_exporter", mode="nice")	0.26
node_cpu_seconds_total(cpu="0", instance="localhost:9100", job="node_exporter", mode="softirq")	8.12
node_cpu_seconds_total(cpu="0", instance="localhost:9100", job="node_exporter", mode="steal")	0.18
node_cpu_seconds_total(cpu="0", instance="localhost:9100", job="node_exporter", mode="system")	175.27
node_cpu_seconds_total(cpu="0", instance="localhost:9100", job="node_exporter", mode="user")	474.39
node_cpu_seconds_total(cpu="1", instance="localhost:9100", job="node_exporter", mode="idle")	6726.26
node_cpu_seconds_total(cpu="1", instance="localhost:9100", job="node_exporter", mode="iowait")	98.77
node_cpu_seconds_total(cpu="1", instance="localhost:9100", job="node_exporter", mode="io")	0
node_cpu_seconds_total(cpu="1", instance="localhost:9100", job="node_exporter", mode="nice")	0.34
node_cpu_seconds_total(cpu="1", instance="localhost:9100", job="node_exporter", mode="softirq")	9
node_cpu_seconds_total(cpu="1", instance="localhost:9100", job="node_exporter", mode="steal")	0.39

total count of cpu usage in seconds from starting

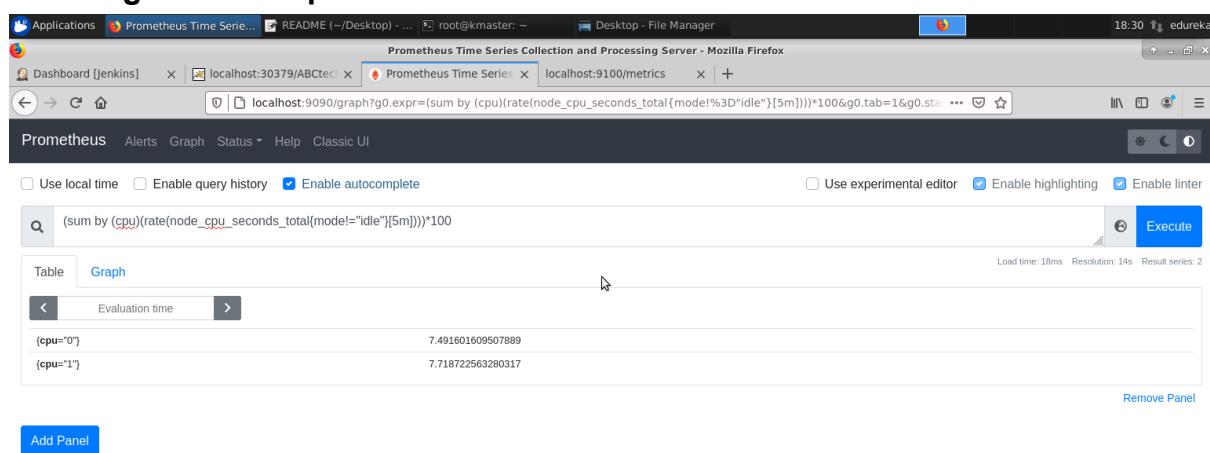
The screenshot shows the Prometheus UI with the query `sum by (cpu)(node_cpu_seconds_total{mode!="idle"})`. The results are displayed in a table:

Series	Value
(cpu="0")	740.64
(cpu="1")	753.5899999999999

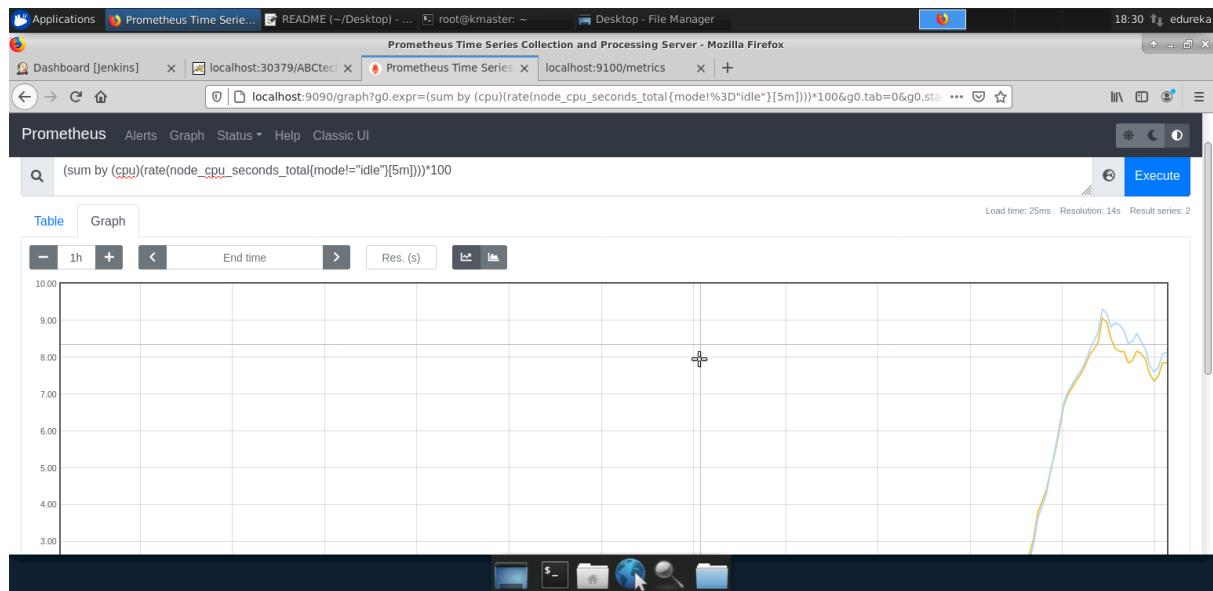
Total count of cpu in seconds from starting



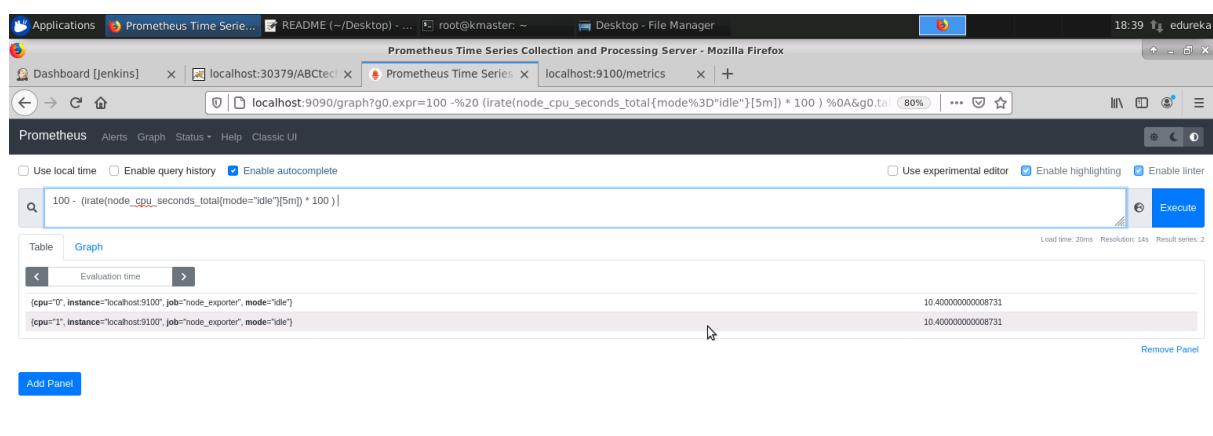
Rate of growth for cpu in last 5 minutes



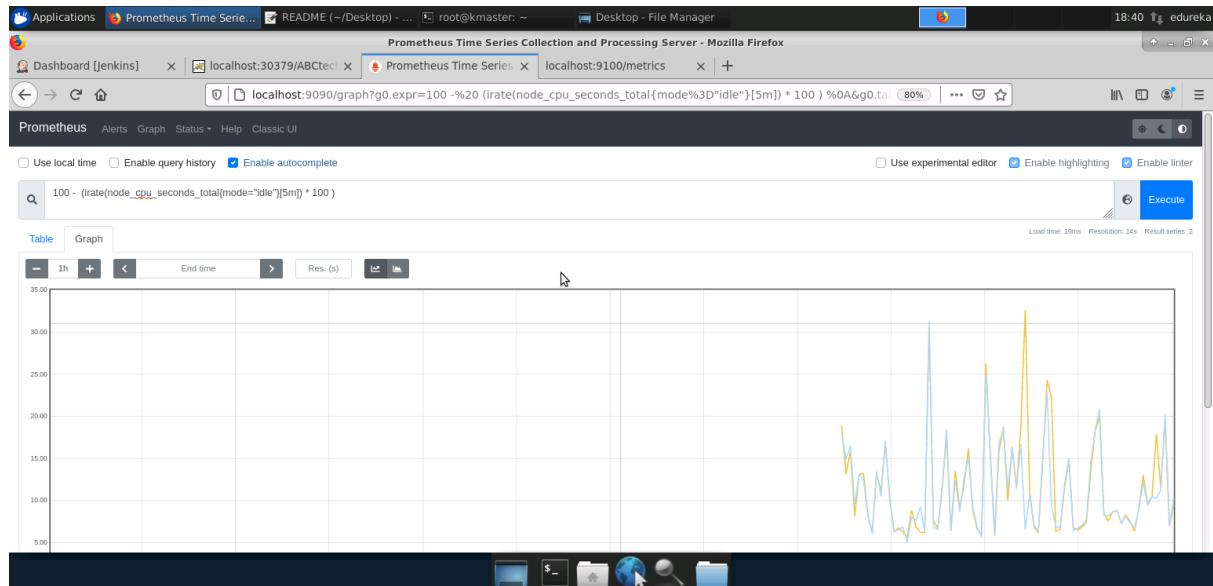
Rate of growth for cpu in 5 minutes graph



CPU Utilization



CPU Utilization graph

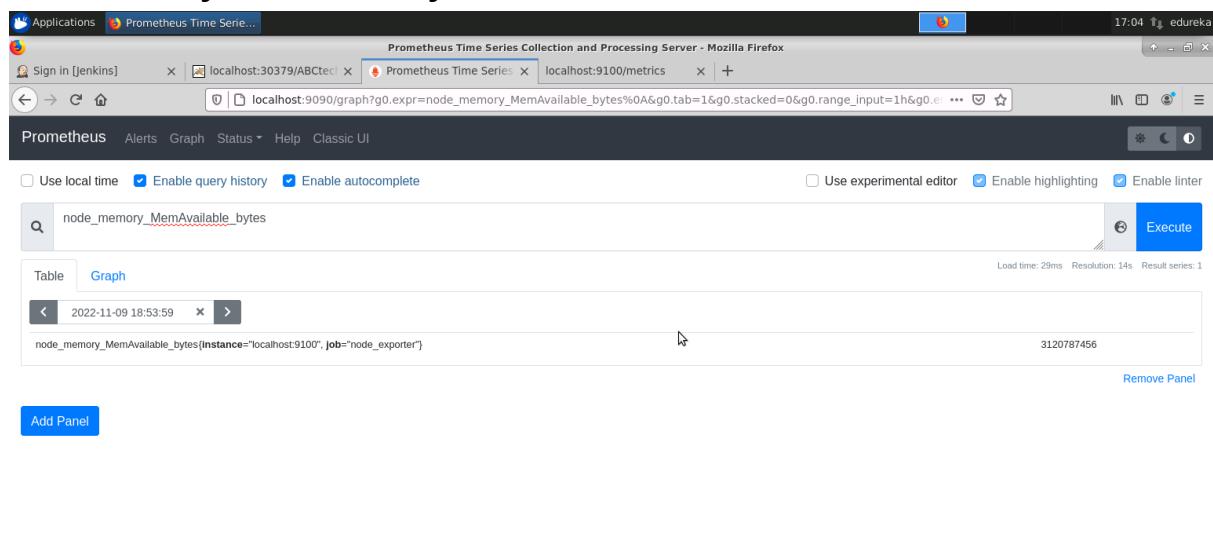


Some of the commands i have used for CPU monitoring

```
sum by (cpu)(node_cpu_seconds_total{mode="idle"})
(sum by (cpu)(rate(node_cpu_seconds_total{mode="idle"}[5m]))*100
100 - (avg by(instance)(node_cpu_seconds_total{mode="idle", instance="localhost:9100"}[5m])) * 100 ) - CPU Utilization;
100 - (avg by(instance)(node_cpu_seconds_total{mode="idle", instance="localhost:9100"}[5m] offset 24h)) * 100 ) - CPU Utilization offset with 24hours ago;
( {1 - rate(node_cpu_seconds_total[job="node-exporter", mode="idle", instance="localhost:9100"])[10m]} / ignoring(cpu) group_left count without (cpu) (node_cpu_seconds_total[job="node-exporter", mode="idle", instance="localhost:9100"]) ) - CPU Utilization per Core;
( {1 - rate(node_cpu_seconds_total[job="node-exporter", mode="idle"])[10m]} / ignoring(cpu) group_left count without (cpu) (node_cpu_seconds_total[job="node-exporter", mode="idle"])) - CPU Utilization per Core;
```

Memory monitoring

Node memory available in bytes:



Node memory available in MB

The screenshot shows the Prometheus UI in Mozilla Firefox. The URL in the address bar is `localhost:9090/graph?g0.expr=node_memory_MemAvailable_bytes%2F1024%2F1024%0A&g0.tab=1&g0.stacked=0&g0.ran...`. The search bar contains `node_memory_MemAvailable_bytes/1024/1024`. The results table shows a single row with the metric `{instance="localhost:9100", job="node_exporter"}` and a value of `2976.21484375`. The UI includes tabs for Table and Graph, and a date range selector from 2022-11-09 18:53:59 to 2022-11-09 18:54:00. There are also checkboxes for local time, query history, autocomplete, experimental editor, highlighting, and linter.

Node memory available in MB for last 5min

The screenshot shows the Prometheus UI in Mozilla Firefox. The URL in the address bar is `localhost:9090/graph?g0.expr=avg_over_time(node_memory_MemAvailable_bytes[5m])/1024/1024`. The search bar contains `avg_over_time(node_memory_MemAvailable_bytes[5m])/1024/1024`. The results table shows a single row with the metric `{instance="localhost:9100", job="node_exporter"}` and a value of `2994.000716145834`. The UI includes tabs for Table and Graph, and a date range selector from 2022-11-09 18:53:59 to 2022-11-09 18:54:00. There are also checkboxes for local time, query history, autocomplete, experimental editor, highlighting, and linter.

memory usage in percentage

The screenshot shows the Prometheus UI in Mozilla Firefox. The URL is `localhost:9090/graph?g0.expr=100 * (1 - ((avg_over_time(node_memory_MemFree_bytes[10m]) + avg_over_time(node_memory_Cached_bytes[10m]) + avg_over_time(node_memory_Buffers_bytes[10m])) / avg_over_time(node_memory_MemTotal_bytes[10m])))`. The result table shows a single row with the value 60.31753584297807.

Series	Value
{instance="localhost:9100", job="node_exporter"}	60.31753584297807

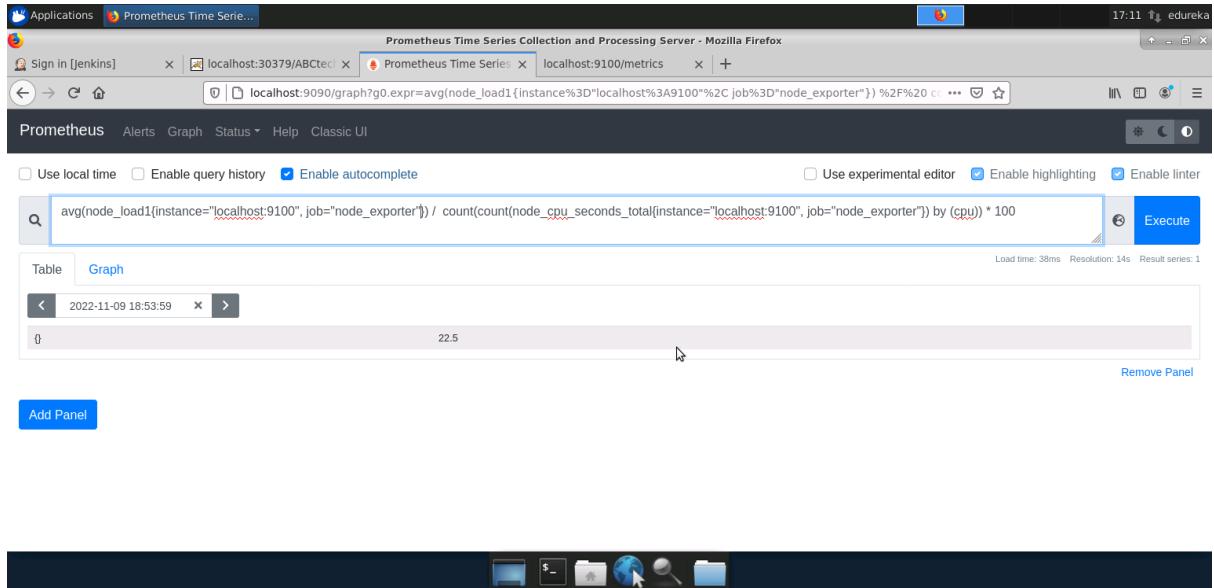
Memory Available by Node

The screenshot shows the Prometheus UI in Mozilla Firefox. The URL is `localhost:9090/graph?g0.expr=node_memory_MemAvailable_bytes * on(instance) group_left(nodename) (node_uname_info)`. The result table shows a single row with the value 3120787456.

Series	Value
{instance="localhost:9100", job="node_exporter", nodename="kmaster"}	3120787456

Network monitoring

Load average in percentage:



Load Average per Instance:

The screenshot shows the Prometheus UI in Mozilla Firefox. The URL is `localhost:9090/graph?g0.expr=sum(node_load5{}) by (instance) / count(node_cpu_seconds_total{mode="user"}) by (instance) * 100`. The results are displayed in a table with one row, showing a value of 25 for the instance `{instance="localhost:9100"}`. The UI includes tabs for Table and Graph, and a time range selector from 2022-11-09 18:53:59 to 2022-11-09 18:53:59.

Network inbound traffic per Node:

The screenshot shows the Prometheus UI in Mozilla Firefox. The URL is `localhost:9090/graph?g0.expr=sum(rate(node_network_receive_bytes_total[1m])) by (device, instance) * on(instance) group_left(nodename) (node_uname_info)`. The results are displayed in a table with seven rows, each representing a network interface (device) and its statistics. The table includes columns for nodename, device, instance, and various byte counts.

nodename	device	instance	bytes received
kmaster	docker0	localhost:9100	0
kmaster	ens5	localhost:9100	1293.3614560794226
kmaster	flannel.1	localhost:9100	0
kmaster	lo	localhost:9100	523637.08929577976
kmaster	vethc1df9a8	localhost:9100	169.70016546357073
kmaster	vethd35395980	localhost:9100	162.26339618524648
kmaster	cnio0	localhost:9100	275.4513882575413

Network outbound traffic per Node

The screenshot shows the Prometheus Time Series Collection and Processing Server interface. A search bar contains the query: `sum(rate(node_network_transmit_bytes_total[1m])) by (device, instance) * on(instance) group_left(nodename) (node_uname_info)`. The results are displayed in a table:

device	instance	nodename	node_uname_info	value
{device="flannel.1", instance="localhost:9100", nodename="kmaster"}		kmaster		0
{device="lo", instance="localhost:9100", nodename="kmaster"}		kmaster		523637.08929577976
{device="venhd1df9ab", instance="localhost:9100", nodename="kmaster"}		kmaster		782.9154317508227
{device="venhd3539580", instance="localhost:9100", nodename="kmaster"}		kmaster		772.2239394876084
{device="cnlo0", instance="localhost:9100", nodename="kmaster"}		kmaster		1555.139371238431
{device="docker0", instance="localhost:9100", nodename="kmaster"}		kmaster		0
{device="ens5", instance="localhost:9100", nodename="kmaster"}		kmaster		13692.965070822045

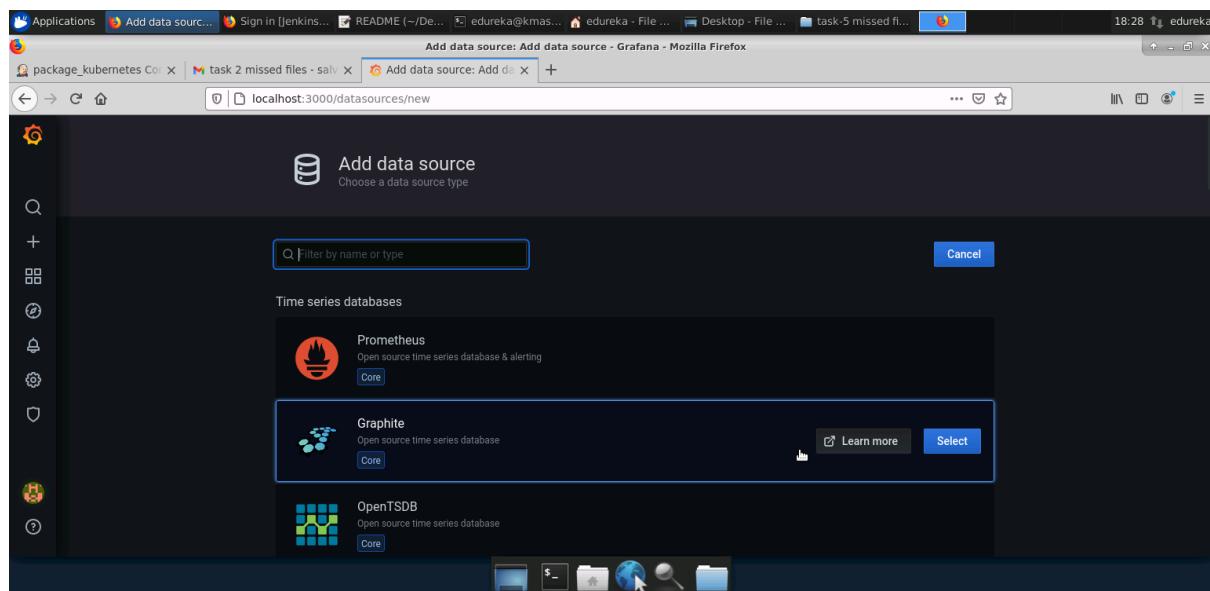
Load time: 44ms Resolution: 14s Result series: 7

Grafana screenshots:

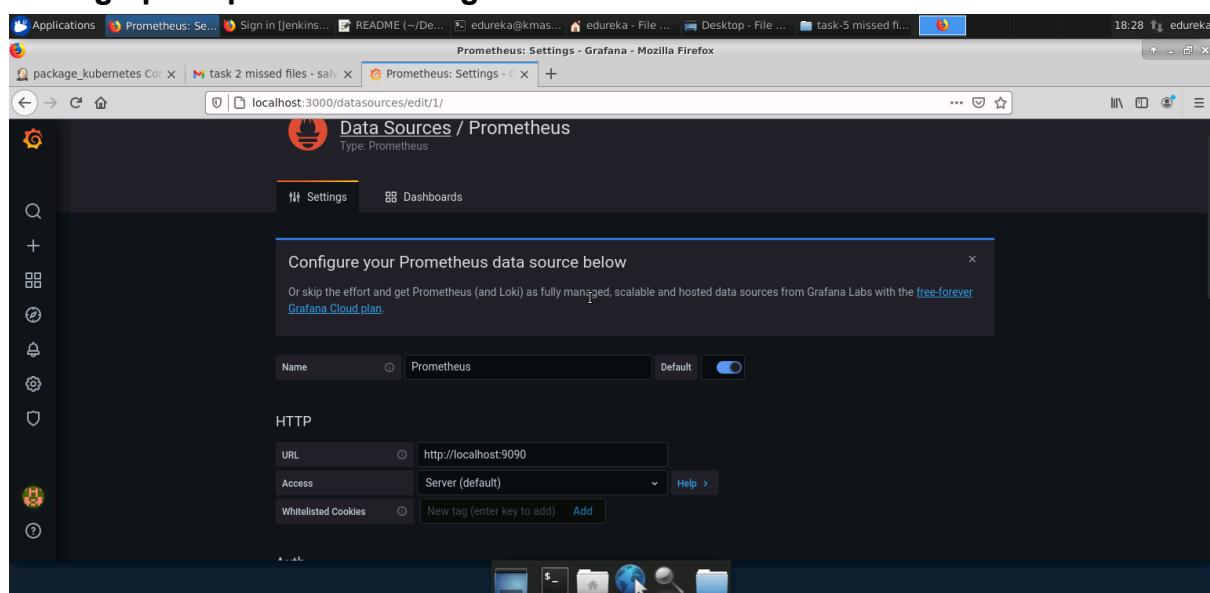
Grafana home page:

The screenshot shows the Grafana home page. The main header says "Welcome to Grafana". On the left, there's a sidebar with icons for General, Dashboards, Metrics, Plugins, and Help. The main content area has sections for "Advanced" (Manage users and teams and add plugins), "TUTORIAL USERS" (Create users and teams), and "PLUGINS" (Find and install plugins). There are also links for "Documentation", "Tutorials", "Community", "Public Slack", and "Latest from the blog".

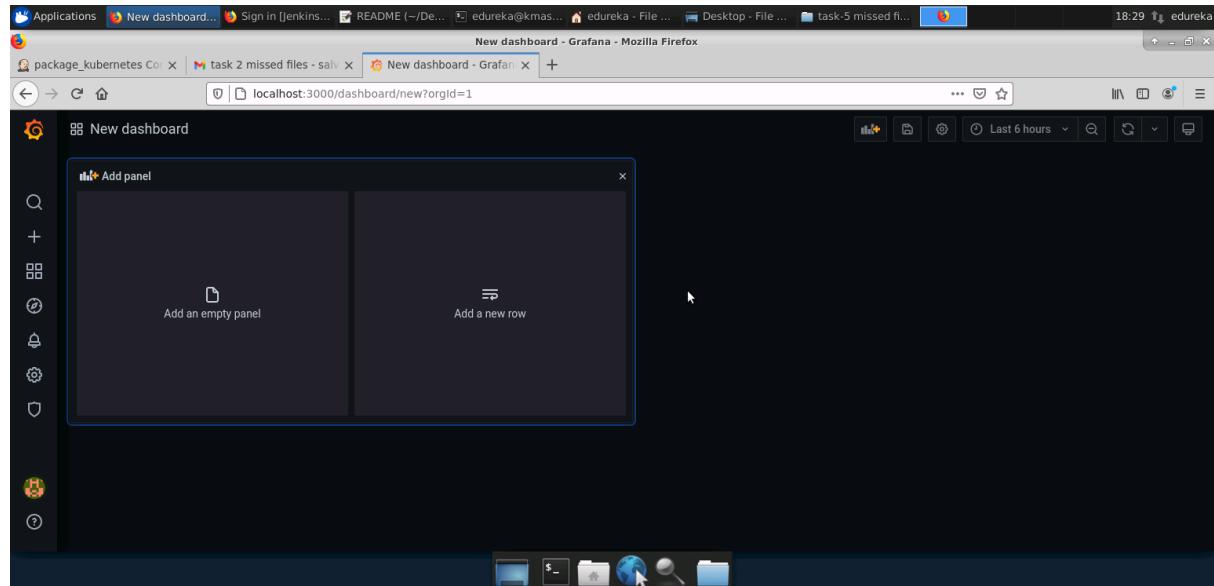
Grafana data source selection(prometheus):



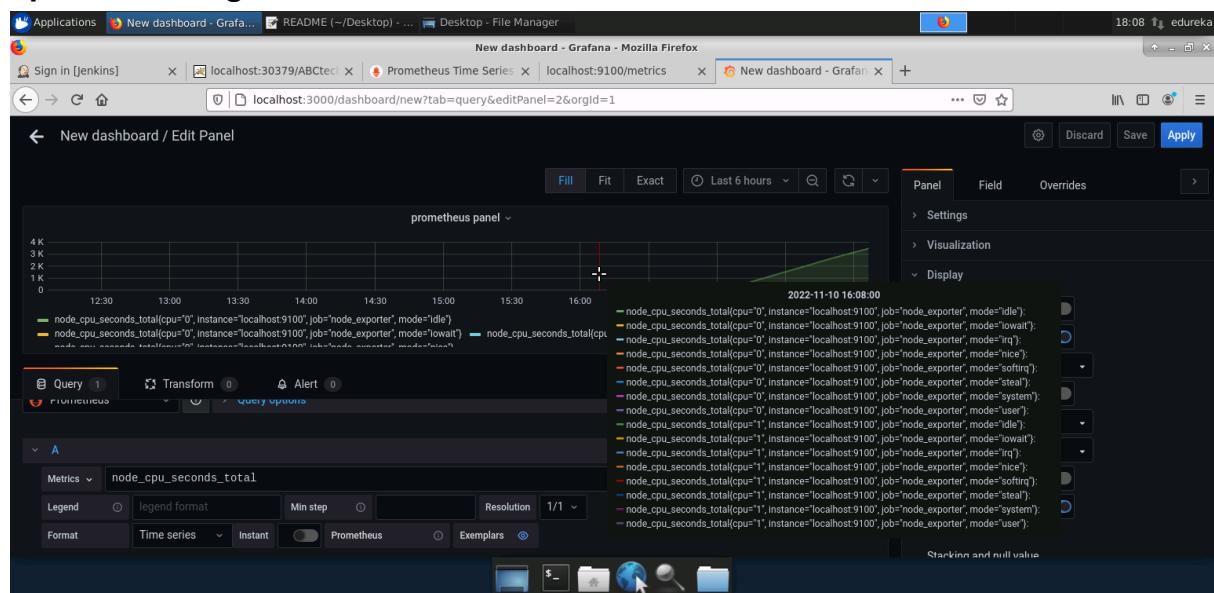
Setting up the prometheus target:



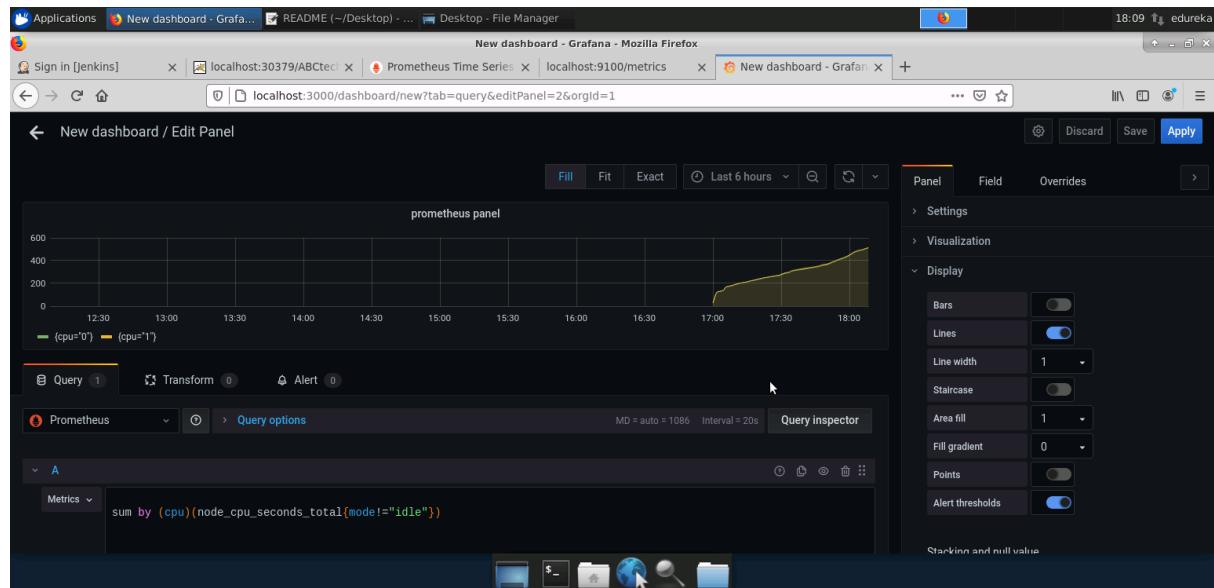
Adding panel for prometheus:



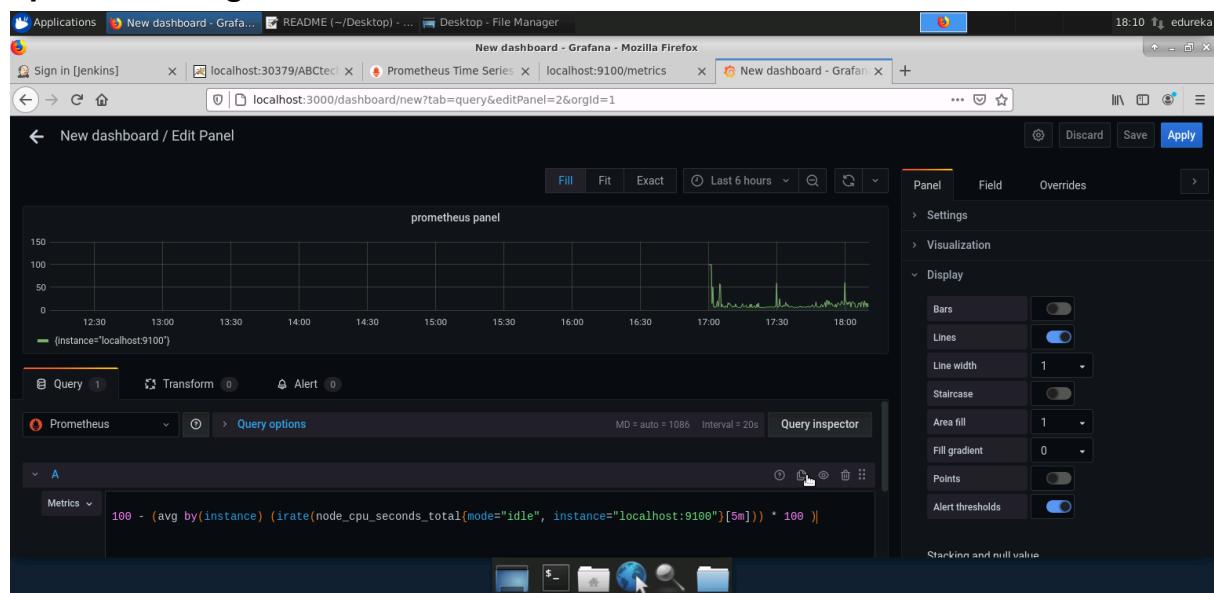
Cpu monitoring 1



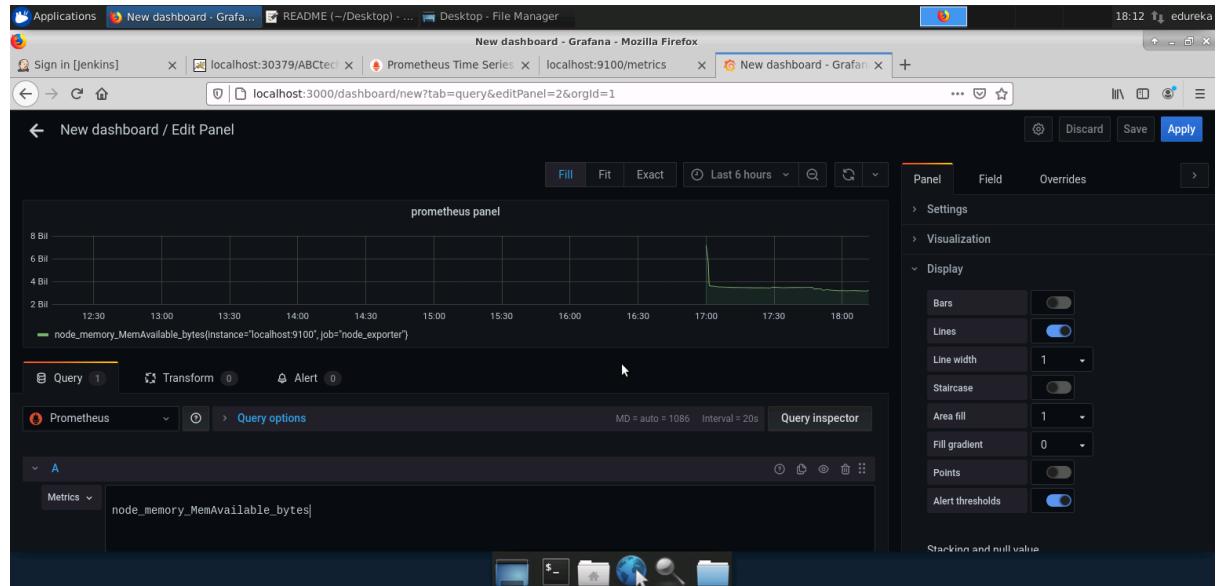
Cpu monitoring 2



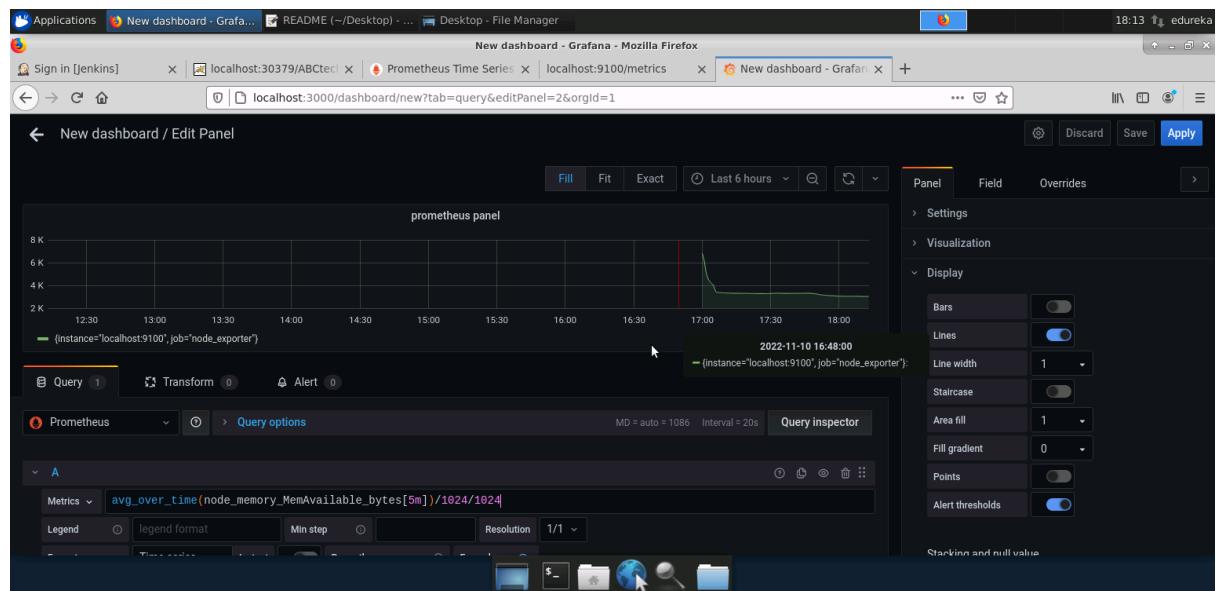
Cpu monitoring 3



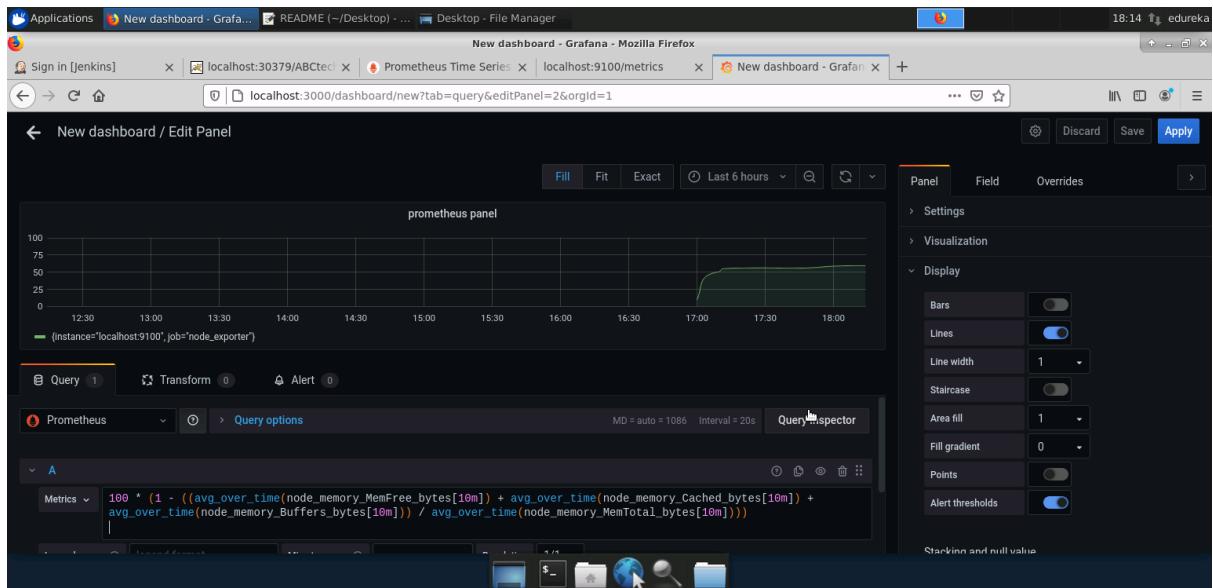
Memory monitoring 1



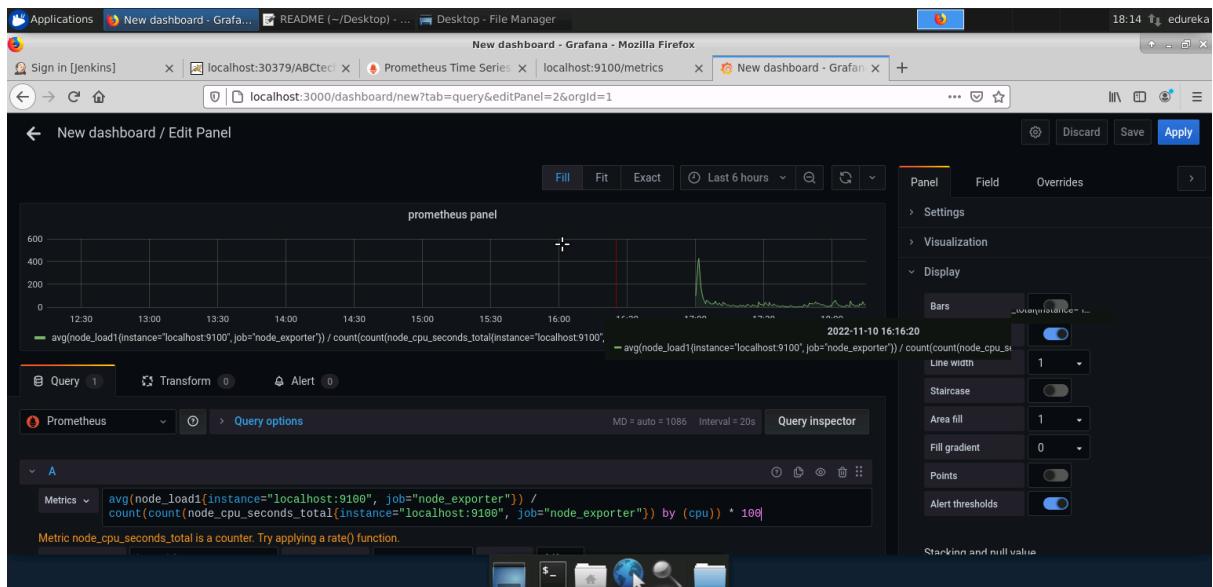
Memory monitoring 2



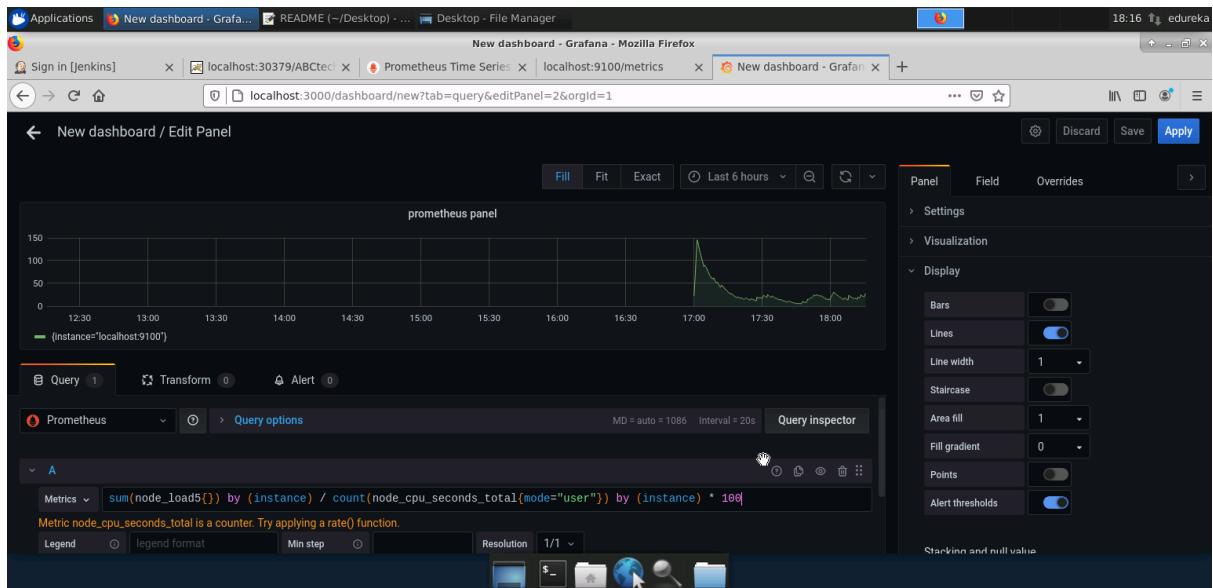
Memory monitoring 3



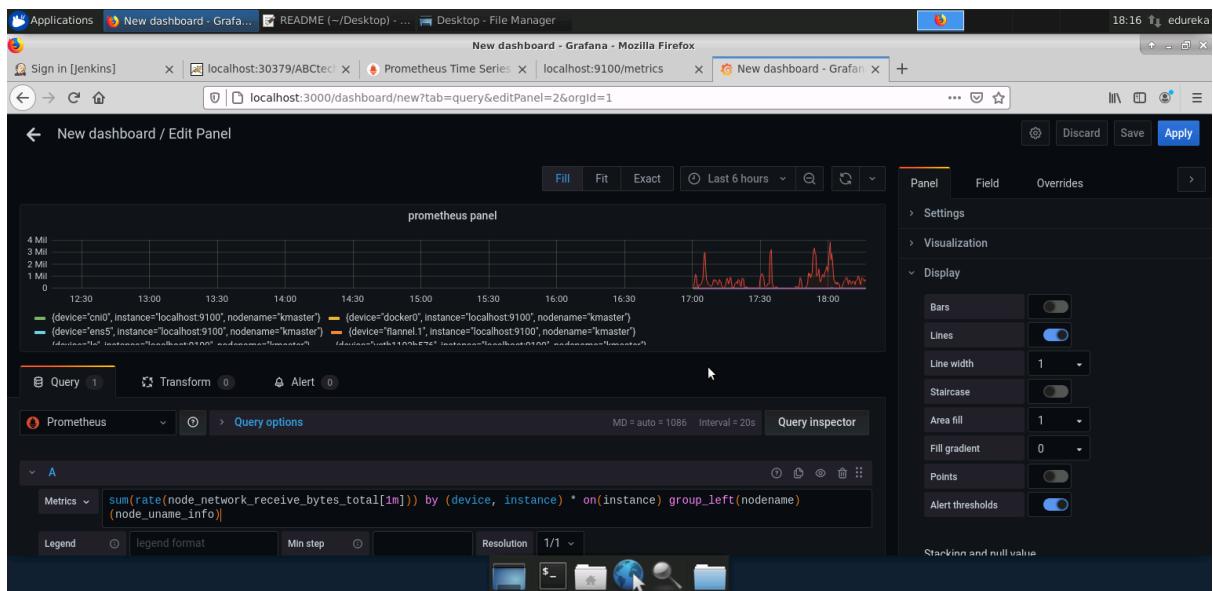
Network load monitoring 1



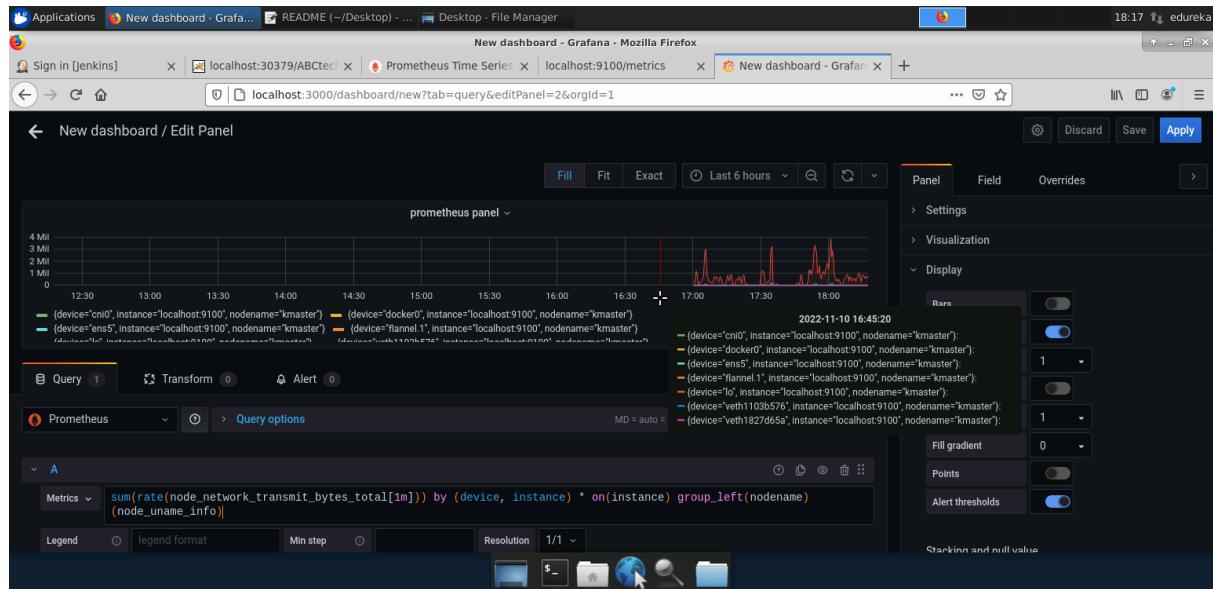
Network load monitoring 2



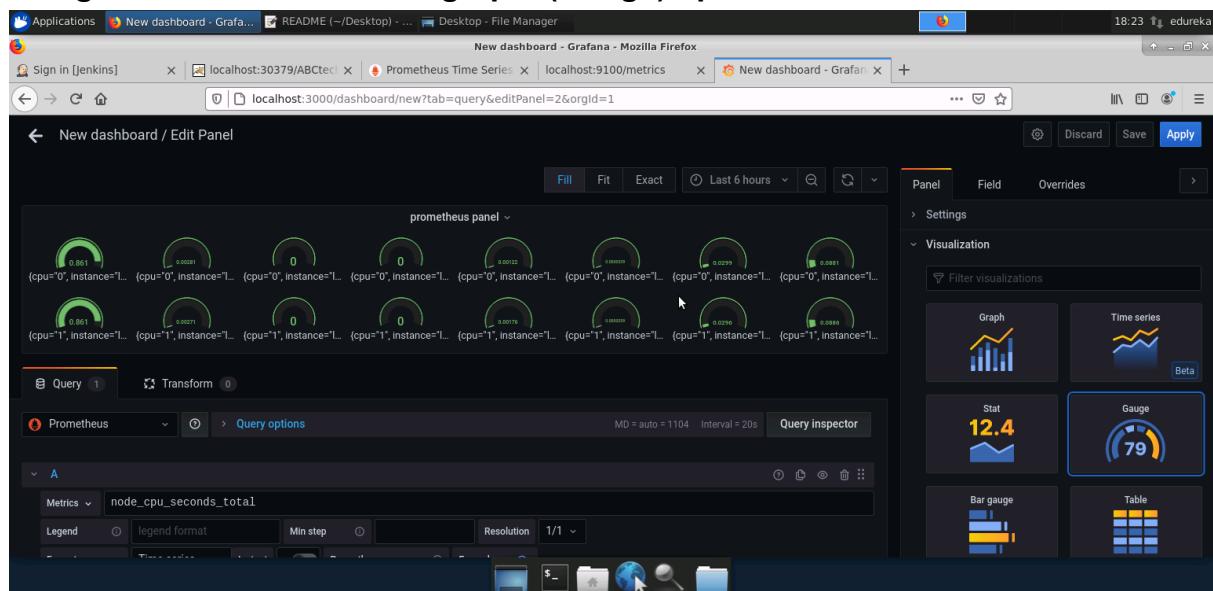
Network monitoring 3



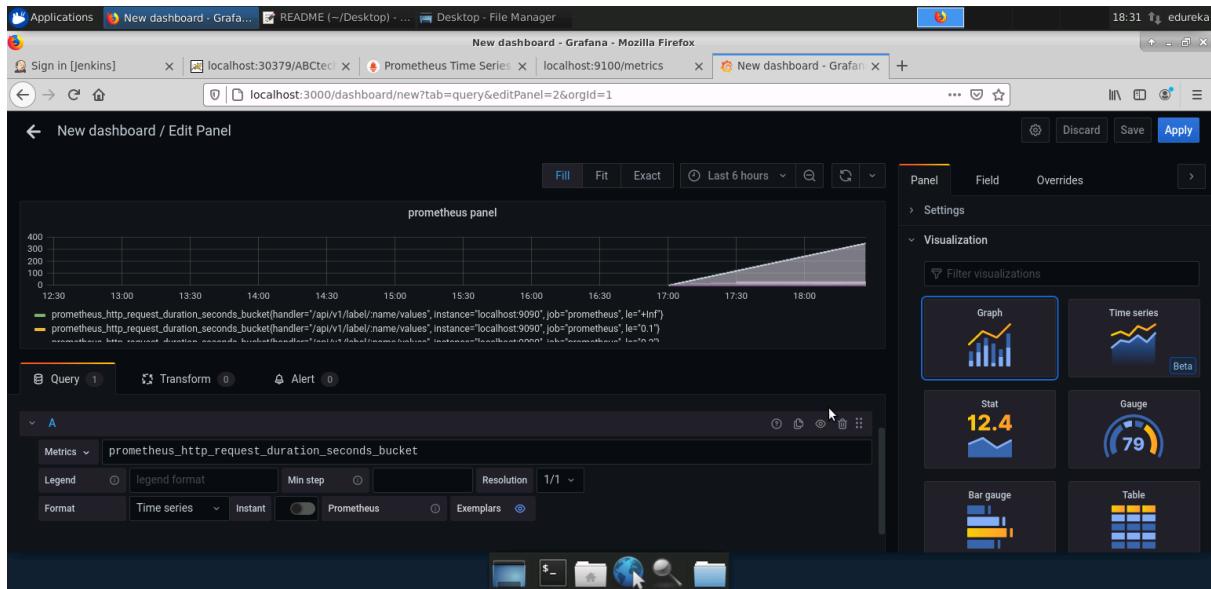
Network load monitoring 4



Using different visualization graphs(Gauge)-cpu breakdown



Using different visualization graphs 2



Commands i have used for prometheus:

```
memory:  
Amount of Memory Available:  
    node_memory_MemAvailable_bytes  
Amount of Memory Available in MB:  
    node_memory_MemAvailable_bytes/1024/1024  
Average Memory Available for Last 5 Minutes:  
    avg_over_time(node_memory_MemAvailable_bytes[5m])/1024/1024  
Memory Available by Node:  
    node_memory_MemAvailable_bytes * on(instance) group_left(nodename) (node_uname_info)  
Memory Usage in Percent:  
    100 * (1 - ((avg_over_time(node_memory_MemFree_bytes[10m]) + avg_over_time(node_memory_Cached_bytes[10m])) / avg_over_time(node_memory_free_bytes[10m])))  
Load Average in percentage:  
    avg(node_load1(instance="localhost:9100", job="node_exporter")) / count(count(node_cpu_seconds_total{instance="localhost:9100", job="node_exporter"}) by (cpu)) * 100  
Load Average per Instance:  
    sum(node_load5()) by (instance) / count(node_cpu_seconds_total{mode="user"}) by (instance) * 100  
Network IO per Node:  
inbound-    sum(rate(node_network_receive_bytes_total[1m])) by (device, instance) * on(instance) group_left(nodename) (node_uname_info)  
outbound-   sum(rate(node_network_transmit_bytes_total[1m])) by (device, instance) * on(instance) group_left(nodename) (node_uname_info)
```

Thank you